PRIJS ƒ 7.25/Bfr. 135

Commodore INFO

Nieuwe Amiga's

Silicon Disk

Actief Activision

ONAFHANKELIJK BLAD VOOR COMMODORE GEBRUIKERS

JAARGANG 4, NO. 3, MEI/JUNI 1987



COMMODORE-INFO

Jaargang 4, no. 3, mei/juni 1987

Uitgave: Sala Communications

Uitgever: Drs. J. Taverne

REDAKTIE

Ir. L. Sala hoofdredacteur
J. Bodzinga adj. hoofdred.
K. van der Vlies reportage
B. Munniksma onderwijs
T. v.d. Land machinetaal
U. Schuurmans software
R. Goudriaan listings

ART DEPARTMENT

Strip: Bert Tier

Illustraties: Ben van Mierlo, Ymmot

Advertentie-exploitatie: Ing. V. Sala, R. Akker

Redaktieadres: Postbus 112, 1260 AC Blaricum, ☎ 02152-65695

Abonnementen en administratie: Postbus 5570, 1007 AN Amsterdam, Den Texstraat 5a, 1017 XW Amsterdam, ☎ 020-273198

Vragen betreffende abonnementen bij voorkeur schriftelijk, met meesturen van het omslagetiket. Telefonisch uitsluitend van 10.00 tot 15.00 uur.

Abonnement: (8 nummers) ƒ 47,50 of Bfr. 950 per jaar

Betaling op Giro 1585491 tnv. SAC/COMMODORE-INFO Blaricum of in België op Bank BBL nr. 310050602562, vermeld SAC/COMMODORE-INFO. Oude nummers à ƒ 6,75 alleen bij vooruitbetaling op een van bovenstaande rekeningen. Ook telefonische opgave voor een abonnement is mogelijk. Bel GRATIS 06-0224222, HP-Teleservice, elke dag tot 20.30 uur (ook in het weekend). Dit nummer is alleen voor telefonische opgave van NIEUWE abonnementen.

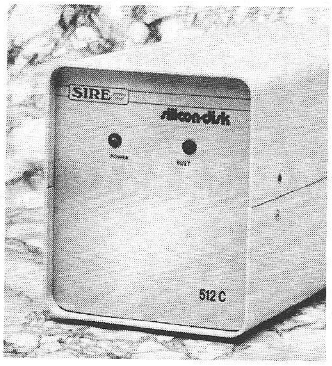
Redactiesecretariaat: Ron van Zalinge

Druk: Verweij, Mijdrecht; NDB, Zoeterwoude

Distributie:
in Nederland Betapress/Gilze
in België AMP/Brussel

ISSN: 0169-3085

Inhoud van dit nummer



PRINT-OUT Listing-rubriek 37

REDACTIONEEL

Commodore leeft! De duizenden, die zich verdrongen rond de Commodore stand op de Computer Info beurs konden zien, dat Commodore met de nieuwe Amiga's weer helemaal in de race is. En wanneer we ons niet vergissen, komt met ook op PC gebied binnenkort met een heel aantrekkelijke low-end machine voor de "instappers" in de PC markt. Men gaat de strijd met de Schneider en andere goedkope klonen niet meer uit de weg.
Dergelijke signalen zijn verheugend, want als makers van een Commodore blad zijn we natuurlijk afhankelijk van wat Commodore zelf doet. Hun nieuwe produkten houden het interessant voor ons, voor de lezers en voor de bedrijven die met hun produkten ook weer afhankelijk van de commodore markt.
Maar men is nu in ieder geval weer heel actief bezig en ook voor de 64 blijft men met nieuwe dingen komen. Wat dat betreft staat Geos nog volop in de belangstelling en ook in dit nummer daaron weer aandacht voor dat Operating Systeem met een speciale rubriek.

L.S.

Het opmaken van teksten met de computer, om zo (met behulp van een laserprinter) nauwelijks van zetwerk te onderscheiden drukwerk te maken, is pas sinds kort levensvatbaar, met behulp van een PC of Apple Macintosh. In korte tijd zijn er ook al vergelijkbare pakketten voor de meeste andere types computer verschenen. Eén van deze paketten voor de Commodore Amiga is Pagesetter.

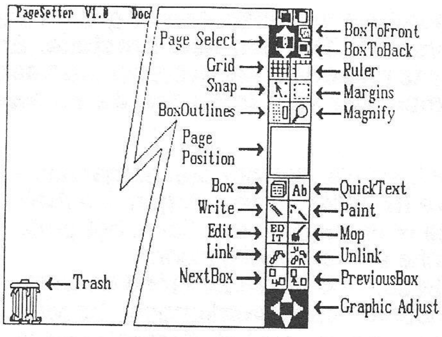
PAGESETTER

Waar je ook kijkt of wie je ook spreekt, iedereen heeft het over DESK-TOP PUBLISHING, kortweg DTP. Het werkt als een toverwoord en voor iedere computer is er wel een programma. Wie had er een jaar geleden van "WYSIWYG" gehoord? Voor de niet ingewijden de betekenis: What You See Is What You Get. Of in goed Nederlands: alles wat er op het beeldscherm staat wordt ook afgedrukt. Ook voor de AMIGA is sinds kort zo'n programma in Nederland verkrijgbaar.

Het programma genaamd Pagesetter is ontwikkeld door de firma Gold Disk uit Canada. Het is een volledig Desk-Top Publishing pakket dat alle mogelijkheden van de Commodore Amiga volledig benut. Met Pagesetter is het mogelijk om tekst en tekeningen te combineren, in een groot aantal stijlen en lettertypes. U kunt het gebruiken voor het samenstellen van bijvoorbeeld een rapport, krant of tijdschrift, maar bijv. een visitekaartje is ook mogelijk. U heeft de volledige controle over de layout en het ontwerp, zodat u uw werk een professioneel uiterlijk kunt geven.

Starten

Na de kickstart, bij alle Amigagebruikers welbekend, is het laden van Pagesetter eenvoudig. De kickstart uit de drive halen, Pagesetter erin en zodra het workbench-scherm opkomt 1 keer klikken op het icon. Het programma wordt dan geladen. Er verschijnt een beeld zoals in figuur 1 is afgebeeld. Het programma is volledig menugestuurd. Als eerste moet u een pagina creëren door de gewenste optie met de muis aan te wijzen. Er verschijnt nu een pagina op het scherm die volledig leeg is.



Figuur1:StartschermvanPagesetter

Boxen plaatsen

U komt nu aan het sterkste punt van Pagesetter, namelijk de mogelijkheid om verschillende boxen op de pagina te zetten. Boxen zijn de sleutel tot desktop publishing, maar vormen in feite niet meer dan een drager voor tekst of een tekening. Bij het plaatsen van de boxen wordt u geholpen doordat de coördinaten op het scherm staan. Ook bestaat de mogelijkheid om over de gehele pagina een raster te zetten. U kunt een aantal boxen koppelen om zo de tekst via de verschillende boxen door te laten lopen. In elke box kunt u weer een nieuwe box zetten. Voor het vullen van een box met tekst zijn er twee mogelijkheden, de eerste is de optie voor het snel invullen van een tekst, die niet langer dan 100 karakters mag zijn. De tweede manier is het gebruiken van de ingebouwde tekstverwerker, die alle gebruikelijke mogelijkheden heeft, maar bovendien teksten uit andere tekstverwerkers voor de Amiga, zoals Textcraft en Scribble, kan inlezen.
Met de grafische editor kunt u tekeningen maken, draaien, spiegelen, vergroten en verkleinen. Zelfs kunt u tekeningen laden uit de diverse grafische paketten zoals De Luxe Paint en Aegis Images.

Configuratie

Wat heeft u nodig om met dit pakket te werken? Een Amiga met minstens 512 K geheugen, en een goede printer, zelfs een matrix printer kan een zeer goed resultaat geven. En natuurlijk niet te vergeten het pakket Pagesetter.

Conclusie

Als u beroepsmatig of privé veel met teksten werkt is Pagesetter een ideaal pakket met veel mogelijkheden, die na een zekere inwerkperiode ook goed te benutten zijn. De prijs van Pagesetter is *ƒ* 480,-

De langzamerhand traditionele voorjaarsbeurs in de RAI, oorspronkelijk begonnen als Commodore-Info Beurs, trok op zaterdag 11 april weer bijna 11.000 bezoekers. Vooral de nieuwe Amiga's van Commodore stonden in het brandpunt van de belangstelling, men verdrong zich rond de Commodore stand.

Computer Info Beurs 11 april

Deze beurzen blijken in een grote behoefte te voorzien, een zo groot aantal bezoekers op één dag is zelfs voor de RAI een hele prestatie. En hoewel de meesten komen om een keus te maken uit het grote aanbod aan hard- en software, bekeek men toch met zeer veel interesse de nieuwe produkten.

De nieuwe Amiga 500 en 2000 beleefden op de Computer Info beurs hun officiële publiekspremiere. Het management van Commodore was dan ook persoonlijk aanwezig om de reakties van het publiek op deze belangrijke gebeurtenis te peilen.

Verbazing

De aantrekkelijk geprijsde en veelzijdige en krachtige Amiga 500 is bedoeld om op termijn de opvolger van de 64 en 128 te worden en men hoopt dan ook op verkopen, die in de tienduizenden moeten gaan lopen. Men wil heel welbewust niet de PC markt van zakelijk en serieuze gebruikers op gaan. De Amiga biedt ook voor wie toch nog de link naar de MS-DOS wereld wil hebben, wel voldoende aanknopingspunten en emulaties.
Bovendien heeft Commodore ook een hele reeks PC's en AT's (modellen 10-20-40) en zijn er ook in die richting plannen om later dit jaar nieuwe modellen uit te brengen. De Amiga is er voor het creatieve, het grafische en het speel-plezier.



De duizenden, die zich rond de Commodore stand verdrongen, hadden soms moeite om het allemaal goed te kunnen zien, de belangstelling was gewoon te groot. De verrassing voor velen was het feit, dat men de Amiga qua prijs nu plotseling wel bereikbaar zag worden; het wat elitaire en dure imago van de machine wordt afgebroken door een prijskaartje van minder dan 1500 gulden inclusief voor de Amiga 500.

Prijsoorlog

Als gebruikelijk was er weer een hele prijsoorlog aan de gang op de beurs. Dat is ook de aanleiding voor heel wat bezoekers om naar deze beurs te komen. Ze willen profiteren van de lage prijzen en de kans, om alle aanbiedingen eens naast elkaar te zien. Pas dan wil men een keus maken, en daarbij wordt gelukkig niet alleen op de prijs gelet, de naam en service van de leveranciers spelen daar wel degelijk een rol bij. Opvallend was eigenlijk de relatief stabiele prijs van de 64. Er waren alleen wat machines uit Australie, die wat beneden de gebruikelijke prijzen weggingen, maar in het algemeen kan men de 64 niet meer goedkoop op de kop tikken. Dat komt omdat de verkoop nog steeds goed loopt en er eigenlijk te weinig 64's zijn. Iedere dealer zou ze wel willen hebben.

Pagesetter

Voor de Amiga werd er op de Hotware-stand (02152-62343) het Desk-Top Publishing programma Pagesetter gedemonstreerd. Dat is een van de eerste bijna professioneel te noemen opmaakprogramma's voor de Amiga. Een ander pakket op dit gebied, Vizawrite, zou door Altycos getoond worden, maar op het laatste moment bleek men uit Engeland de verkeerde versie gekregen te hebben, heel jammer.

PC arena

Wat wel duidelijk werd op deze beurs, is de overgang naar de PC door heel wat "educated users". De freaks en hackers stappen in grote aantallen over op de MS-DOS computers. Voor rond de tweeduizend gulden kan men tenslotte al een "kloon" aanschaffen, al is die voor die prijs vaak wat mager uitgerust. De taiwanese klonen van verschillende huizen met allemaal leuke fantasienamen deden het op de beurs goed, al zag men toch ook erg veel kopers kiezen voor de Amstrad en andere merk-klonen.
Het is erg moeilijk te beoordelen, of een goedkope PC nu ook werkelijk zal voldoen. Het vreemde is, dat de compatibiliteit, vaak de grootste zorg van de koper, bij de allergoedkoopste modellen meestal beter is dan bij de iets duurdere modellen. Dat komt omdat de taiwanese makers daarvan ge-



woon de IBM PC compleet kopiëren, zonder op mogelijke inbreuk of legale complicaties te letten.

> **Het kopen van een goede IBM-kloon blijft voor velen een moeilijke keuze, waarbij een beurs als deze veel informatie kan leveren**

Hun klonen zjn dan ook vrijwel gelijk aan de IBM qua BIOS en andere besturings-aspecten. Juist de iets duurdere systemen, waar men de moeite doet om een eigen legale BIOS te gebruiken, falen daarin iets vaker. De keerzijde van deze medaille is, dat de duurdere modellen ook betere componenten en subsystemen gebruiken. En er zijn bijvoorbeeld diskdrives te koop (op fabrikantenprijsniveau), die niet meer dan een gulden of 80 kosten, terwijl de meer betrouwbare modellen meer dan het dubbele kosten. De kwaliteit is navenant: we kennen heel wat bezitters van klonen, die binnen de kortste keren met onleesbare en niet te redden bestanden zitten. En ook met de voedingen, connectors en printeraansluitingen is het niet alles goud wat er blinkt.

"Educated users"

Het blijkt, dat bladen als Commodore-Info en PC Business Info toch voornamelijk een publiek bereiken, dat we zouden kunnen aanduiden als "*educated users*". Dat zijn computerbezitters of gebruikers, die goed geïnformeerd zijn over wat er te koop is, weten wat ze willen, en zoeken naar voordelige aanbiedingen en nieuwtjes. Daarvoor lezen ze computerbladen, bezoeken beurzen en gaan naar gebruikersgroep-bijeenkomsten. Die "bewuste gebruiker" is enerzijds een lastige klant voor de leverancier, hij weet er vaak meer van dan de verkoper, vraagt veel en wil maar een lage prijs betalen, aan de andere kant heeft hij een zeer duidelijke voortrekkersfunctie. Eén zo'n gebruiker kan vele andere, minder deskundige potentiele kopers in zijn omgeving over de streep trekken. Dat gaat niet altijd zo duidelijk in zijn werk, maar het feit dat meneer A of B zelf merk Z gebruikt, beïnvloedt veel anderen, die dat toevallig zien of horen.

Naar de PC

Opvallend was ook op deze beurs de verschuiving van hobby- naar MS-DOS computers. Een steeds groter aantal hobbyisten heeft -na een leerschool op de 64 of vergelijkbare machine- nu het oog laten vallen op de kleinere PC's.

Daarmee wordt dan ook wel geëxperimenteerd, maar vaak worden er ook wel meer serieuze toepassingen aangepakt. datacommunicatie en werken met bulletin-boards is ook bij de PC's de grote mode. Wat dat betreft liet deze beurs ook zien dat de tendens naar de PC met veel communicatiemogelijkheden nog een geweldige toekomst tegemoet gaat. Modems in veel (goedkope) soorten en maten waren daar deze keer al een voorbode van. Natuurlijk herbergde deze beurs ook veel kleine en grote handelaren, die eens lekker hadden uitgepakt met goedkope voordeeltjes op allerlei terrein (diskettes, opbergdozen, uiteenlopende software etc.)

DTP Shoot Out

Nieuw op de beurs was de Shoot Out voor DeskTop Pub lishing pakketten. In Amerika al lange tijd een gebruikelijke vorm van vergelijking tussen pakketten, maar aan deze kant van de grote plas nog nooit eerder gehouden. Het recept is eenvoudig: een aantal opdrachten zijn op floppydisk gezet en worden uitgereikt aan een groep ervaren DTPgebruikers, die in een zo kort mogelijke tijd een zo goed mogelijk resultaat proberen te ontwerpen. De opdrachten zijn meestal een brief, een groter document (eventueel met graphics), en zaken als ingewikkelde headers en footers, gecompliceerde opmaakopdrachten etc. Ook deze Shoot Out was goed voorbereid. Een achttal redactionele opmakers uit onze computerbladenwereld waren bijeengezet om de klus te gaan klaren. Eigenlijk stond niets een lange en spannende race in de weg...

Het enige waaraan de organisatie (die prima in elkaar zat) niet op gerekend had, was de enorme interesse en aandacht die het publiek had voor de nieuwe pakketten, die aan deze Shoot Out meededen! Veel deelnemers wilden wel opdrachten verwerken, maar vonden voortdurend geïnteresseerde bezoekers aan hun tafel, die wilden weten hoe het betreffende pakket werkte en wat de voor en nadelen van de diverse pakketten waren. Vooral de wat uitgebreide programma's zoals PageMaker (voor de MSDOS machines!) en Ventura scoorden hoog in belangstelling.

Het gevolg was een lange dag voor de acht opmakers, die hadden gedacht een aantal opdrachten relatief rustig te kunnen uitvoeren.

Met schorre kelen, en knap vermoeid snakte deze werkers naar het koele pilsje dat aan het eind van de dag op hen stond te wachten. Maar ook voldaan, dat een groot publiek kennis heeft kunnen nemen van hun pakketten en de mogelijkheden daarvan.

Succes

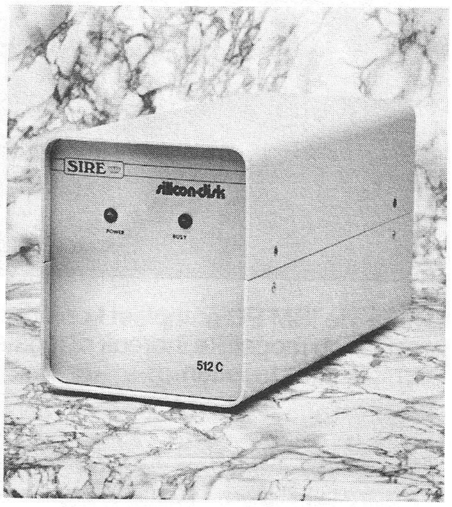
Een algemene conclusie te trekken over deze dag is niet zo lastig. Een grote hoeveelheid mensen heeft zich te goed kunnen doen aan nieuwe en goedkope hard en software, leuke en belangrijke informatie en enig zicht gekregen op de onstuimige ontwikkelingen, die deze branche ongetwijfeld nog enige jaren zal blijven doormaken.

Een succes voor iedereen, die op het gebied van de eigen hobby bij wil blijven. Maar ook voor diegene, die steeds meer naar de toekomst kijkt, waarin een zakelijk gebruik van de als hobby aangeschafte apparatuur bij onontkoombaar is.

●

Opslag van gegevens is een belangrijk aspect van het computergebruik. Veel Commodore-gebruikers werken met een data-recorder of diskdrive, maar er is nu ook een zeer geavanceerd opslagmedium voor de Commodore beschikbaar.

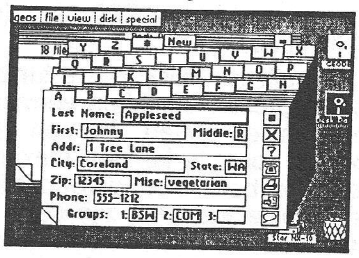
SILICON DISK



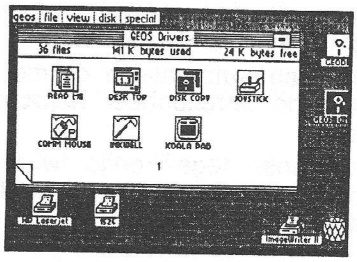
Door de firma RECOMPUT uit Rotterdam wordt een nieuw opslagmedium op de markt gebracht, dat werkt met de snelheid van het licht.

De silicon disk gaat uit van een nieuw concept: de statische Ram. Het transporteert data met de snelheid van het licht, geruisloos en zonder fouten. Er bevinden zich geen bewegende delen in het apparaat, dit in tegenstelling tot andere opslagmogelijkheden. De firma SIRE uit Dongen heeft de disk ontwikkeld voor de IBM-compatible machines, voor de Commodore 64 en 128 is dit apparaat aangepast door RECOMPUT.
De gebruiksmogelijkheden moeten worden gezocht, daar waar elke andere drive het veilig transporteren van gegevens snel zou opgeven. Denk hierbij vooral aan stoffige ruimtes en aan werksituaties waar magnetische velden voorkomen. Een extra reden waarom deze disk interessant kan worden is de geweldige snelheid waarmee gewerkt wordt. Het wachten tot een programma geladen is is niet een van de plezierigste bezigheden. Hiervan kunnen de 1541 gebruikers meepraten.
De silicon disk kan worden aangesloten op een Commodore 64 en 128 en is dan geheel compatible met de 1541 diskdrive. Hij wordt aangesloten op de expansiepoort en neemt geen geheugen van de computer in beslag. De software die nodig is voor het gebruik zit in een cartridge. Na het aansluiten en inschakelen van de disk herkent de computer hem als disk met een device 10. Het devicenummer kan softwarematig worden aangepast door poke 827,n waarbij de waarde van n kan variëren van 9 tot 40. De 1541 kan gewoon met het device 8 worden aangesloten. Het laden van een programma van bijvoorbeeld 160 blokken duurt maar 4 (!!) seconden. De silicondisk wordt standaard geleverd in een 512 K uitvoering. Er zijn uitbreidings modules verkrijgbaar waarbij u telkens 512 K erbij plaatst. Het geheugen van de silicon disk wordt na het uitschakelen van de computer door een batterijtje gevoed. Bij het niet gebruiken van de disk blijven de files minimaal twee jaar bewaard. Als u de computer regelmatig gebruikt wordt de spanning minimaal 6 jaar gewaarborgd.
Er kunnen maximaal 248 verschillende programma's of files worden opgeslagen. De opslagcapaciteit is standaard 2048 blokken waarvan er 2009 zijn te gebruiken. Het apparaat is geheel ongevoelig voor netstoringen en schokken. Alle standaard leverbare software draait probleemloos op de disk. Vooral met sequentiële files is de snelheid verbluffend.
Zijn er dan helemaal geen nadelen aan deze silicon disk? Zoals elke nieuwe vinding heeft ook deze zijn kinderziektes of aanloopmoeilijkheden. De disk die wij ter recensie ontvingen werkte niet op de 128. Een handleiding ontbrak volledig bij onze disk. Van de leverancier hebben we vernomen dat de handleiding is nagestuurd naar alle gebruikers en nu meegeleverd wordt. Voor de Commodore 128 wordt er een nieuwe Rom meegeleverd zodat er probleemloos mee kan worden gewerkt zowel in de 128 als de 64 mode. Er wordt op dit moment gewerkt aan software speciaal voor gebruik met deze disk.
Adviesprijs voor de standaard uitvoering is *f* 999,- De uitbreiding per 512 K kost *f* 750,- ●

Op de computerbeurzen in Las Vegas en kort geleden in Hannover was Berkely Softworks ruim vertegenwoordigd. Hand in hand met Commodore zelf, die GEOS tot het officiële DOS heeft uitgeroepen, exposeerde men de nieuwste snufjes.



GEOS rubriek



In de vorige nummers bespraken we al Geodex, Deskpack, Fontpack en Writer's Workshop, maar Berkeley Softworks heeft nog meer in petto! Men introduceerde Geofile, een database en Geocalc, de spreadsheet. In de Berkeley Softworks catalogus staat bovendien ook Deskpack II aangekondigd en een GEOS-muis, daar zullen we later nog op terugkomen.

De Berkeley Softworks pakketten zien er allemaal goed verzorgd uit, al zijn ze tamelijk duur en zijn de uitbreidingen in ons land niet overal verkrijgbaar.

Geofile

De kenmerken van dit databaseprogramma kunnen kort worden weergegeven:

° De data worden geplaatst in 'forms', die door de gebruiker bepaald kunnen worden. (een form kan bijvoorbeeld 'naam' of plaats heten.). Bij PC-databases wordt een form wel 'record' genoemd, en de onderdelen waar dat record (veld) uit bestaat, de 'tags'.

° Geofile beschikt over een krachtig, variabel zoek-commando

° Er kunnen adres-labels geprint worden

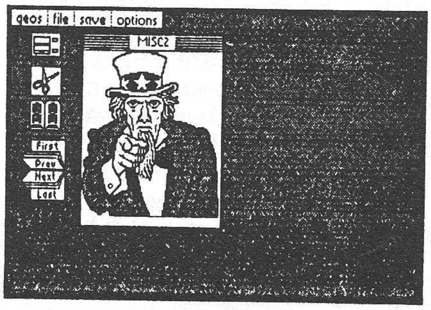
° De gegevens worden automatisch gesorteerd.

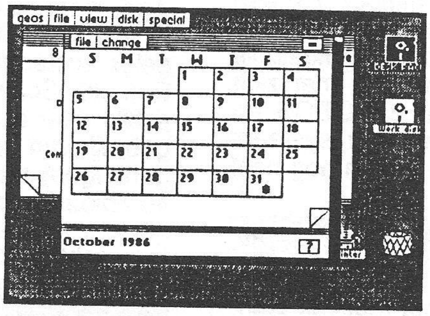
Nu doemt natuurlijk de vraag op: waar zijn twee databases in een systeem nu goed voor? Volgens Berkeley omdat het twee totaal verschillende systemen zijn. Het verschil tussen Geodex en Geofile is, dat Geodex voornamelijk bedoeld is om namen en adressen in op te slaan, alleen voor gebruik in mailmerge- brieven. De opzet van Geodex is ook veel kleinschaliger, dat blijkt wel uit het feit dat het gehele programma constant in een overlegschermpje staat. Geofile daarentegen, beslaat constant een gehele pagina. Deze volwaardige database is veel uitgebreider en heeft ook veel meer mogenlijkheden wat betreft databasetaal. (de verzameling commando's die ter bescikking staan om een bestand te ordenen.) Met Geofile kan men inventarislijsten, investeringen of postzegelverzamelingen bijhouden. Geodex is meer bedoeld voor het maken van telefoonklappers en adresbestandjes.

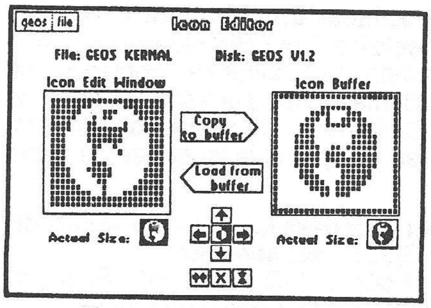
Wij trekken echter in twijfel of het ontwerpen van twee databases voor eenzelfde systeem wel zo verstandig is. Het antwoord op de vraag met welke van de twee u het beste af bent, hangt af van het doel waar u hem voor wilt gebruiken. De prijs van Geofile is ongeveer *f* 125,-

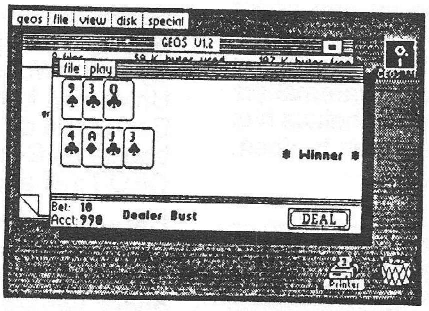
Geocalc

Ook nieuw is de spreadsheet Geocalc, geheel in GEOS stijl, dus met muis, fonts en trekmenu's. Deze spreadsheet, ofwel rekenmatrix, is vooral gericht op persoonlijke financiële analyse, zoals budgetten regelen, kosten schatten, de belasting berekenen en voor het maken van (eenvoudige) financiële 'modellen'. Met Geocalc kan men niet alleen kijken









De schermopbouw in de Geosprogramma's is altijd zeer overzichtelijk.

naar bestaande toestanden, maar ook mogelijke situaties met de functie 'What If' doornemen.

De belangrijkste kenmerken van Geocalc zijn:

° 112 kolommen en 256 rijen (dus 28.000 cellen)

° et gebruik van wiskundige functies zoals optellen aftrekken vermenigvuldigen en delen. In een eenvoudige formule kunnen maximaal 12 cellen betrokken worden, bij ingewikkelde formules zijn dit 9 cellen.

° De grootte van cellen kan door de gebruiker veranderd worden.

° Men kan automatisch of met de hand de spreadsheet herberekenen.

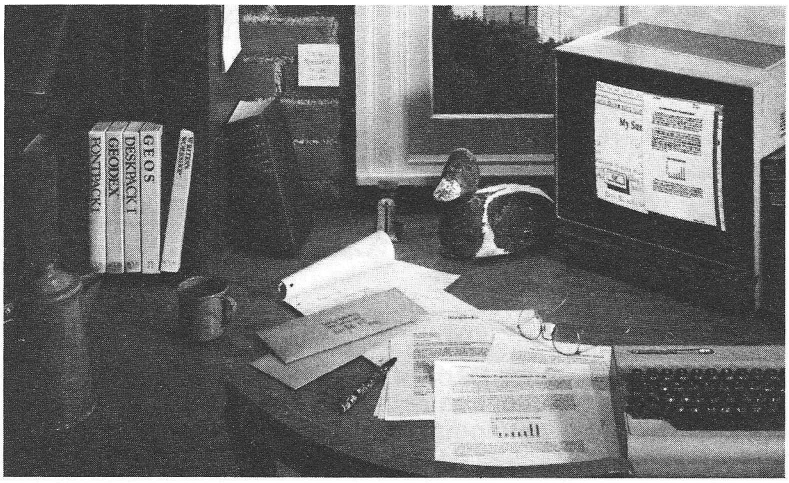
° Er kunnen tegelijkertijd twee gedeeltes van de spreadsheet in beeld zijn.

°

Al met al een aantrekkelijk programma dat het beroemde drietal tekstverwerker-database-spreadsheet nu ook bij GEOS compleet maakt. De prijs zal in Nederland rond de *f* 125,- komen te liggen.

WRITER'S WORKSHOP™

THE COMPLETE WRITING ENVIRONMENT FOR USE WITH GEOS™



GeoWrite is een van de toppers uit het GEOS-pakket.

Vragen en antwoorden bij GEOS

Wanneer zal de C 64C met geïntegreerde GEOS, dus zonder laden, verkrijgbaar zijn?

Voorzover ons bekend laat GEOS zich niet zomaar in een ROM-pack duwen. Daarvoor zouden allerlei laadroutines en opslagmethodes gewijzigd moeten worden. Omdat het BOOTEN van GEOS minder dan een minuut vergt, vinden wij dat GEOS op ROM niet dringend gewenst is. Een uitgebreide C-64c zou de kostprijs omhoogdrijven en compatibiliteitsproblemen met zich meebrengen. Dit betekent echter niet dat GEOS op ROM nooit verwezenlijkt zal worden, het systeem staat nu pas in de kinderschoenen en er kan nog heel wat gebeuren.

Is het mogelijk is om met Geowrite op een AZERTY toetsenbord te werken?

Normale Commodore 64's zijn uitgerust met het standaard Qwerty toetsenbord. Een andere, veel minder gebruikte standaard is AZERTY. Als je nu een ander toetsenbord aan je '64 hebt gezet omdat dat lekker tikt kunnen er wel eens problemen ontstaan. Slechts in weinig gevallen (zoals Vizawrite) spelen softwaremakers op dergelijke problemen in en het is dan ook helaas niet mogelijk om wat aan deze Geowrite perikelen te doen, als u niet goed thuis bent in machinetaal.

Ik heb problemen met het kopieren van files en diskettes en met het werken met diskettes in het algemeen. Geos zegt me nooit, dat ik van disk moet wisselen. Wat doe ik fout?

Geos onderscheidt diskettes alleen qua naam. Als u bij bijv. kopiëren van disk wisselt en die disk heeft dezelfde naam als de vorige, dan kan er veel schade aangericht worden. Geos gaat dan de files op zichzelf kopiëren en niet op de doel-disk. Went u zich dus aan, de disk bij het formatteren al een eenmalig voorkomende naam te geven; als uw disk al geïnstalleerd is, kunt u de naam in het DISK menu met 'rename' veranderen.

Wanneer zal GEOS ook voor de 128 verkrijgbaar zijn?

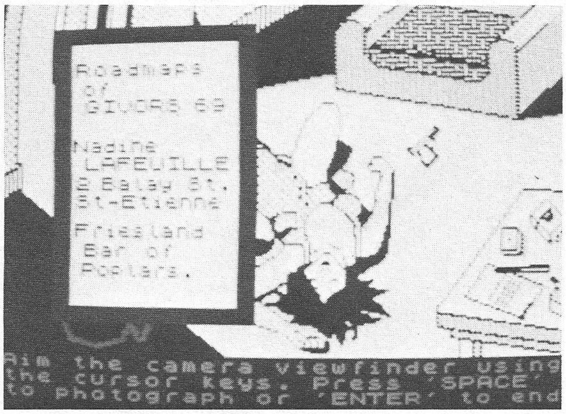
Geos voor de C-128 werd al een tijd geleden aangekondigd, maar zoals zo vaak gebeurt in de computerwereld, is nog niet uitgebracht. Tot nu toe behoort een dergelijk systeem dus tot de 'vapourware'. Er wordt bij Berkeley Softworks echter wel aan gewerkt en men beschikt daar al over een 128 versie. Deze versie maakt gebruik van de faciliteiten die de 128 biedt (een groter geheugen) maar de grafische tekeningen zullen in de 40-koloms modus komen te staan, aangezien de 80-koloms veel te langzaam werkt. Voorlopig blijft het echter wachten tot een goede versie in de handel komt.

Hoe komt men eigenlijk aan de namen in het GEOS systeem?

GEOS komt van **Graphical Environment Operating System**. En bijvoorbeeld het font BSW staat voor Berkeley SoftWorks. Namen als California, Fontknox, en University komen allemaal van plaatsen in en rond Berkely (in de staat Californie). De wereldbol (het symbooltje bij Geowrite en Geopaint) is weer van de naam GEOS zelf afgeleid.

Dat was het dan weer voor deze keer. Blijf ze sturen, die vragen en tips voor GEOS. Het adres is:
Commodore Info, Postbus 112, 1260 AC Blaricum, onder vermelding van: 'GEOS' gebruikersrubriek'.

Thrilling Commodore



Misdaadadventures

De misdaad slaat de laatste tijd weer hard toe bij de C-64 en C-128. Nee, het gaat niet om hackers of computerdieven, maar om thrillers waar de liefhebbers hun bebloede vingers bij af kunnen likken. Adventures waartegen de oude "cops en robbers"arcadegames verbleken en de spanning ten top stijgt. Dit alles dankzij een sterke plot en grote realiteit. We bekeken twee van deze Commodore-thrillers voor u.

Vera Cruz

Voor de Hercule Poirots onder de Commodore-bezitters brengt **Infogrames** (via Ariolasoft 64,-) de adventure-thriller Vera Cruz uit. Als brigadier-rechercheur uit Saint Etienne (Frankrijk) kan de speler op zijn dooie gemak een heel intrigerende moord oplossen. Een telefoontje haalt de held uit de formidabele Franse cuisine en brengt hem naar het appartementenblok Forest waar één van de bewoners (?) Vera Cruz dood op de grond ligt.

Het moordwapen, een pistool, ligt naast haar en alles duidt op zelfmoord of toch niet? Wellicht brengt de informatie van de huisbewaarder de brave brigadier een stukje verder.

Er zijn twee spelschermen. De eerste is natuurlijk de plaats van het misdrijf kompleet met slachtoffer, pistool, bloed en tal van verborgen aanwijzingen. Alle bewijsstukken, o.a. de handtas en het dagboek, worden met behulp van de cursor gefotografeerd voor de latere bewijsvoering.

Als een goed speurder moet de speler echt minitieus te werk gaan anders worden belangrijke aanwijzingen gemist. Gebruik die cursor als een stofkam en sla geen plekje over! Daarna kan de detective de verklaringen, via de S van statements op het menu, opnemen. Er zijn spelfiguren genoeg, maar of die allemaal wat zinnigs te zeggen hebben moet blijken.

Gewapend met al het bewijsmateriaal drukt de speler op ENTER en het tweede spelscherm verschijnt, een gecomputeriseerd politiebureau.

Eerst worden alle gegevens netjes in volgorde op de schrijfmachine (die functioneert als itemlijst) gezet en weer met een ENTER ingevoerd. Daarna kunnen de verworvenheden van de moderne politietechnieken via het gecomputeriseerde politienetwerk benut worden. De speler heeft hiermee toegang tot een enorm informatiearsenaal.

Vond u bijvoorbeeld het kenteken van een auto dan verschaft de computer de nijvere speurder naam en adres van de eigenaar. Met de namen van die personen kan de brigadier bij andere politiebureau's naar hun dossiers vragen.

Het aardige van Vera Cruz is de grote realiteitszin. Geen flitsende arcadeacties of zo maar eens wat standaard adventure-termen intypen. Alles draait, ondanks de moderne voorzieningen, om nauwkeurig en langzaam voortschrijdend speurwerk.

En gemakkelijk is Vera Cruz beslist niet. Relevante informatie wordt zo gemist en vele sporen lopen dood. Des te groter is echter de voldoening als de ware killer bij de kladden wordt gepakt.

Moonmist

HELP roept uw vriendin Tamara in die laatste brief. Tamara staat doodsangsten uit want niet alleen schijnt iemand haar te willen vermoorden, maar er is ook nog sprake van een witte dame in de nacht, een zwarte weduwe (de giftige spin) in bed en een vervaarlijke slang in de bureaulade. Het speurdersbloed begint natuurlijk direkt in de aderen te borrelen. Hier is geen sprake van ovoorzichtige terrariumhouders, maar er hangt moord in de lucht.

Fluks afgereisd dan maar naar het oude Tresylian familiekasteel van vriendin Tamara in Cornwall. Het avontuur begint voor de kasteelpoort waar de detective zich in mannelijke of vrouwelijke persoon (svp opgeven) kan aandienen. Een en ander is uiteraard bepalend voor de verdere rolverdeling, garderobe en uitrusting. Stel, u meldt zich als vrouw aan: dan komt de feodale Lord Jack naar buiten en kust de speurster met Engelse hoffelijkheid de hand. Om de "couleur locale" nog wat te versterken mag de speler zich zelf ook Lady of Baron laten noemen. Een ander persoonlijk tintje is dat de favoriete kleur van de speler aan de Commodore door gegeven kan worden waarop de computer kleding en interieur naar smaak aanpast.

Zo ontspint zich een fraai gothisch mysterie met vele verrassingen en doordachte wendingen. De scenery is sterk sfeertekenend. De volle maan rijst boven de verweerde toren, golven slaan tegen de ruwe klippen. En dan zijn er nog tal van buitenissige personages zoals de Luitenant, Montague Hyde en Vivien Penreath.

Geen omgeving waarin een Amerikaanse **Infocom**-speurneus gauw verzeild raakt. Om het allemaal nog wat intrigerender te maken is Lord Jack verloofd met Tamara die zijn vorige geliefde, Deidre, aan de dijk heeft gezet. Daarop is Deirdre in zee verdwenen of misschien toch niet? Wie spookt er 's nachts als witte dame rond? Gaat het om een bovennatuurlijke macht, wraak of de in het kasteel verborgen schat? Vier verschillende scenario's en een krachtige tekstparser zullen het de speler leren. Geen fraaie graphics maar pure ouderwetse tekstspanning goed voor heel wat avondjes plezier. En voor de beginnende avonturier niet al te moeilijk.

U.S.

Al jarenlang is de schaakcomputer of het schaakprogramma voor uw eigen homecomputer een vertrouwd beeld en onlangs is er ook eindelijk voor de damliefhebbers een kwalitatief goed programma op de markt gekomen.

Dammen,

maar dan met een tegenstander van niveau?

Settle Light Soft, het softwarelabel van het Terneuzense bedrijf Nederlands Handels Instituut (N.H.I.) is de uitgever van dit nieuwe damprogramma.

Settle Light Soft's Dammen is niet echt het eerste damprogramma wat op de nederlands markt te koop is.

Kampioen

Voor de Commodore 64 was er bijv. al eerder "Dammen 64" van Softworld en ook uit Frankrijk zagen we al eerder damprogramma's. Maar de meeste damprogramma's voldeden nauwelijks aan de verwachtingen. Daar is nu verandering in gekomen en dat werd in februari dan ook duidelijk gemaakt.

Settle Light Soft's Dammen werd officieus Nederlands Kampioen. Deze onder auspicien van de K.N.D.B. gehouden wedstrijddag voor computers, was als voorbereiding op het officiele kampioenschap in april natuurlijk al een belangrijk graadmeter en werd glansrijk door het door ons geteste programma gewonnen.

Wat te denken van tegenstanders als de ATARI ST en de APPLE MACINTOSH, beiden uitgerust met de krachtige 68000 microprocessor. Deze moesten tegen Settle Light Soft's Dammen voor de Commodore 64 het loodje leggen.

Spelregelkennis.

Er zijn een aantal zaken belangrijk in dit soort programma's. Zo klinkt het misschien gek, maar we vinden kennis van de spelregels heel belangrijk. Daar blijkt het bij nogal wat programma's aan te schorten. Settle Light Soft's Dammen blijkt goed op de hoogte te zijn met de spelregels, te denken valt aan voorrangsregels bij meerslag, dam over gewone steen, achteruitslaan, kortom alles zit erin. Ook de beginner komt helemaal aan zijn trekken want er zitten maar liefst 200 oefenstanden in, die duidelijk maken hoe in een bepaalde speelsituatie de oplossing moet worden bereikt. Is er een spelsituatie ontstaan die moeilijk of onoverzichtelijk is dan kan zelfs aan de computer gevraagd worden welke zet hij zou adviseren, prompt wordt dan deze zet voor u uitgevoerd. Wanneer u vindt dat een eigen bedachte zet beter is kunt u de computer corrigeren, of dit beter is moet dan blijken...

Spel- en denkniveaus

Ook leerzaam is de mogelijkheid van dit programma om de computer geheel zelfstandig partijen (dus tegen zichzelf) te laten spelen waarbij u op elk gewenst moment zelf naar zwart of wit kunt overschakelen. Niveau's, dus moeilijkheidsgraden van het spel, zijn ook hier natuurlijk aanwezig en totaal zijn dat er maar liefst zeven. Logischerwijze wordt bij het hoger zetten van het niveau de denktijd van de computer ook langer. Om het verhaal ingewikkelder te maken zijn er behalve de 7 denkniveau's ook nog 5 spelniveau's. Deze spelniveau's zijn er om te assisteren bij de 200 oefenopgaven die in het programma aanwezig zijn.

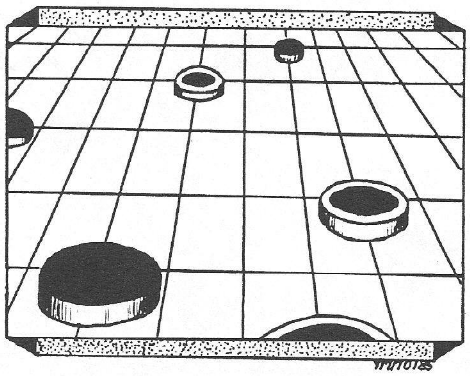
Het spelen zelf, dus het invoeren van de zetten kan op drie manieren gebeuren, nl. met de joystick, met de cursortoetsen of door het intikken van de officiele notatie. Als u spijt heeft van een door u gedane zet kunt u deze terugnemen, de computer blijft sportief en zal dit nooit doen!

Bent u gevorderd of clubspeler dan is een interessante mogelijkheid het opzetten van standen, waarna u zelf met de computer het spel kunt uitspelen, een leuke en leerzame methode.

Conclusie

We hebben gedurende enkele dagen het programma intensief beproefd en vinden dat het programma een goed en uitgebalanceerd geheel is. Qua sterkte prima, en datzelfde kunnen we zeggen van de graphics die erg rustgevend overkomen. Andere leuke opties van het programma zijn het kunnen verwisselen van speelkleur, het spiegelen van de stand (dit is dus niet hetzelfde), het tussentijds kunnen saven en laden van standen zowel voor disk als tape en het automatisch uit laten voeren van een verplichte slag door alleen op de returntoets te drukken.

Bij het programma treft u een helder geschreven handleiding aan, waarin al uw vragen beantwoord worden. De handleiding wordt gecomplementeerd met tactische tips voor het positie- en eindspel. We zouden zonder problemen nog veel meer kunnen schrijven over dit kwaliteitsprodukt van eigen bodem, maar damliefhebbers kunnen het beste gewoon dit damprogramma kopen, want bij de ware dammer mag dit programma niet in de softwarebibliotheek ontbreken.



SOFT WIR WAR

DOOR BERT TIER.



Bert Tier '86.

Basic Min

Een rubriek van Nico Baaijens me

In de vorige Info stond geen Miniatuurtjesrubriek. Die is helaas ergens tussen de redactionele of postale wielen geraakt. Jammer van de aardige programmaatjes van inzenders, die nu ook zoek zijn. Wie denkt dat zijn inzending daardoor ook is zoek geraakt, wordt verzocht zijn bijdrage(n) nogmaals in te zenden.

Overigens blijven de inzendingen gelukkig binnenstromen. Voor mij ligt weer een pak met prima ingezonden materiaal waaruit ik een keuze moet maken.

Laten we met iets nuttigs beginnen. Een stukje machinetaal. Schrik niet want machinetaalprogrammaatjes kunnen ook vanuit een BASIC-programma in memory worden gePOKEt. Dat doet Martijn Oostdijk uit Nijmegen met onderstaand miniatuurtje dat een veel moderner karakterset op het scherm geeft.

```
100 FORI=49152TO49242
110 READX:POKEI,X:NEXT:SYS49152
120 DATA 169,48,133,52,133,56,169,0,141,14,220,169
130 DATA 51,133,1,160,0,185,0,208,153,0,48,200
140 DATA 192,0,208,245,238,19,192,238,22,192,174,19
150 DATA 192,224,224,208,232,169,55,133,1,169,1,141
160 DATA14,220,169,208,141,19,192,169,48,141,22,192
170 DATA 162,0,160,0,169,0,153,2,48,153,4,48
180 DATA 200,200,200,200,200,200,200,200,192,248,
    208, 238
190  DATA 169,29,141,24,208,96,234
```

Alleen voor C128

Blijven we nog even bij de nuttige programma's. Hier komt een minatuur van J. Cremers te Brunssum. Het programmaatje maakt een screendump (een afdruk van een grafisch beeldscherm) op de printer. De (matrix)printer moet dan wel grafisch kunnen afdrukken. Het programma werkt volgens de opgave alleen voor de Commodore 128.

```
100 PRINTCHR$(147)
110 OPEN 4,4
120  BS$=CHR$(8):DE$=CHR$(15)
140  A$="":X=8504
145 FOR K=1 TO 40
150 FOR J=0 TO 24
160 FOR I=0 TO 7
170  AD=X+1
180 C=PEEK(AD)
190 C=255-C
195  IF C<128 THEN C=C+128
200  IF C=>128 THEN C=C
210  A$=A$+CHR$(C)
220 NEXT I
230  X=X+320
240 NEXT J
250 PRINT#4,BS$;A$;DE$
260  A$="": X=X-8008
270 NEXT K
280 CLOSE 4
```

Nog een voor de 128. Een klein maar fijn grafisch miniatuurtje van Roy de Vink uit Loenen a/d Vecht:

```
5  COLOR 0,1:COLOR 4,3
10 GRAPHIC 1,1
20  FOR I=1 TO 90
30 CIRCLE 1,150,100,I
40 NEXT I
```

Het miniatuurtje van Martin Reumers uit Noorbeek kon ik niet testen. Het bevat een looptekst voor C-16 en +4. Kijk zelf maar wat het doet:

```
10  SCNCLR:B$="<39 SPATIES>"
20 INPUT"TEXT ";A$:A$=B$+A$
30  INPUT"SPEED ";CV:A$=A$+"<39 SPATIES>"
40 SCNCLR:L=LEN(A$)-39: FORQ=1TOL: CHAR1,0,0,
   MID$(A$,Q,40)
50 FORT=1TOCV: NEXT: NEXT
```

Vanzelfsprekend ontbreken onze Belgische Commodore-vrienden dit keer niet op het appel. Marc van Loocke uit Gent maakte een intro voor de C-64 en geeft daar gelijk een demootje van in zijn miniatuurtje. Tegelijk geeft Marc ook aan hoe listings kunnen worden ontdaan van grafische tekens, waar we met de tekstverwerker zo veel moeite mee hebben. Laat dit goede voorbeeld goed doen volgen.

```
40  POKE53280,0:POKE53281,0
50  B$="EEN INTRO"   :B=LEN(B$):LB=(40-B)/2
60  C$="VOOR"        :C=LEN(C$): LC=(40-C)/2
70  D$="COMMODORE INFO":D=LEN(D$):LD=(40-D)/2
80 PRINT"(CLR) (WHT) (DOWN) (DOWN) (DOWN)
   (DOWN) (DOWN) (DOWN)"
90 FORA=1TOB:FORX=35TOLB+ASTEP-1:
   PRINTTAB(X-2)MID$(B$,A,1);" (UP)":NEXTX:NEXTA
100 PRINT"(DOWN)(DOWN)(DOWN)"
110 FORA=1TOC:FOR X=35TOLC+ASTEP-1:
   PRINTTAB(X-2)MID$(C$,A,1);" (UP)":NEXTX:NEXTA
120 PRINT"(DOWN) (DOWN) (DOWN)"
130 FORA=1TOD:FORX=35TOLD+ASTEP-1:
   PRINTTAB(X-2)MID$(D$,A,1);"(UP)":NEXTX:NEXTA
```

Nog een intro

Een tikje aan de lange kant voor een miniatuurtje, maar toch erg aardig is de intro (een titelrol) van Ronald den Ottelander uit Dordrecht. Met dit programma komen de tekens als het ware het scherm binnenrollen. Wellicht nuttig voor smalfilm- en video-amateurs.

```
10 PRINT"(CLR)(BLU)  ";:POKE53280,0:POKE53281,0
20 FORT=28672TO28870:READA: POKET,A:NEXTT
   :SYS7*4096
30 INPUT"GEEF EEN TEKST (MAX 25 TEKENS):
(DOWN) (DOWN) (DOWN)(DOWN)  ";A$
40PRINT"(CLR)":IFLEN(A$)>25THENGOTO30
50POKE53272,(PEEK(53272)AND240)+12
60  K=4:HO=30:FORD=1TOLEN(A$)
70 K$=MID$(A$,D,1):LT=ASC(K$)+32*K:FORS=0TOHO
80 POKE1543-S,LT:POKE1544-S,32: LT=LT-32:IFLT<0
   THEN LT=LT+128
90 FOR H=1TO30:NEXTH:NEXTS:HO=HO-1:
   K=K-1:IFK<0 THEN K=3
100 NEXTD:POKE53272,21
110 DATA 162,0,169,0,157,0,48,232,208, 248,238,6,
    112,173,6,112
120 DATA 201,53,208,236,173,14,220,41,254,141,14,
```

iatuurtjes

‹orte tot zeer korte programma's.

```
   220,173,1,0,41
130 DATA 251,141,1,0,162,0,189,8,208,157,8,48,
   232,208,247,173
140 DATA 1,0,9,4,141,1,0,173,14,220,9,1,141,14,220,
   169
150 DATA 0,141,210,112,169,0,141,209,112,169,0,
   141,211,112,169,128
160 DATA 141,208,112,160,1,162,0,189,8,48,45,208,
   112,240,7,152
170 DATA 109,8,49,141,8,49,24,152,42,168,232,224,8,
   208,232,238
180 DATA 211,112,173,211,112,201,8,240,17,24,173,
   208,112,106,141,208
190 DATA 112,238,97,112,238,100,112,76,83,112,238,
   209,112,173,209,112
200 DATA 201,32,240,17,238,97,112,238,100,112,169,
   8,109,88,112,141
210 DATA 88,112,76,73,112,238,210,112,173,210,112,
   201, 3,240,23,169
220 DATA 8,141,88,112,238,97,112,238,98,112,238,
   100, 112,238,101,112
230 DATA 238,89,112,76,68,112,96
```

'Muziek'

Zin in een stukje beatmuziek uit de computer? Dan is er onderstaand miniatuurtje van W. Hogenaar uit Westenschouwen :

```
5 FORX=0TO28:POKE54272+X,0:NEXT
10  V=54296:W=54276:A=54277: H=54273:L=54272
20 FORX=15TO10STEP-1:POKEV,X:POKEA,15: POKE
  H,40: POKEL,200:NEXT
30 POKEW,0:POKEA,0
40 POKEW,129
60 GOTO20
```

Graphics voor de C-16

De C-16 wordt zo gemakkelijk vergeten in de Miniatuurtjes, klagen velen. Misschien maakt het grafische programmaatje van Remco van Dijk uit Alphen aan den Rijn dat een beetje goed.

```
10  A=65:C=0
20 GRAPHIC 1,1
30 COLOR 4,1:COLOR 0,1:COLOR 1,2
40 FOR X=1 TO 320 STEP 1.2
50 Y=INT(100+A*SIN(X/20))
60 DRAW 1,0,100 TO X,Y
70 NEXT X
80  A=-A:C=C+1
90 IF C=2 THEN 110
100 GOTO 30
```

Vreemd

Wat Jan-Adriaan van der Voort uit Zevenbergen allemaal met het scherm van de C-64 uithaalt, is mij een raadsel. Het lijkt wel of hij vreemde grafische effecten bereikt door regelrecht naar bugs in het hardwaresysteem toe te programmeren. Zijn miniprogrammaatje is wel leuk, maar aan het nut kan getwijfeld worden. Door op de Restore-toets te drukken verandert het zonderlinge bewegende beeld. Ga eens kijken!

```
20 FORN=0TO14:READA: POKE49152+N,A: NEXT:
   SYS49152
30 DATA 120,173,17,208,41,239,141,17,208,238,32,
   208,76,9,192
```

Reactie

Twee reactiespelletjes voor de joystick (in poort 2) van Roel Nikkessen uit Bergen (NB) vond ik niet onaardig:

```
10 PRINT"****"
20 PRINT"OPGELET..."
30 FORN=1TO(RND(1)*2000):NEXTN
40 FORA=1TO40
50 PRINTA
60 IFPEEK(56,20)=111THENNEXT:RUN
```

```
10 PRINTCHR$(147)
20 FORN=0TORND(1)*5000:NEXT
30  POKE1000*RND(1)+1024,81:TI$="000000"
40 IFPEEK(197)=64THEN40
50 PRINTINT(TI*5/3)/100
```

SBASIC

Simons BASIC (SBASIC) wint nog steeds aan populariteit. We kunnen dat merken aan het toenemend aantal inzendingen van miniatuurtjes, dat exclusief in Simons BASIC is geschreven. Frank Scharff uit Vianen maakte een leuke, die een balkengrafiek geeft door toevalswaarden:

```
1 DATA 6,5,13,7,8,2,4,14:FORX=0TO7:READK(X):NEXT
   :HIRES 7,0  :DIMY(25):FORX=1TO25
2 Y(X)=20:TEXT 0,(X-1)*8, RIGHT$(STR$(X),2),1,1,8:
   NEXT :REPEAT   :X=INT(RND (0)*25)+1
3 Y(X)=Y(X)+1: LOW COL K(XAND7),0,0:LINEY(X),(X-
   1)*8,Y(X),X*8 -1: UNTIL Y(X)=319
4 WAIT 198,255
```

Voor SBASIC stuurden Eric en Carlo van Dongen, toevallig woonachtig in het Brabantse Dongen, een heel pak SBASIC-miniatuurtjes op. Hoe meer programma's, hoe groter de kans op plaatsing, dachten ze. Ze hebben gelijk. We geven er twee: Wind surfing en ellipsen:

```
10 HIRES 0,1:MULTI 2,7,6
11 FORC=1TO60
20  A=A+1:B=A+1
30 :ANGL 65,B+A,A,B,A,1
40 NEXT C
50 CIRCLE 85,120,48,10,3
60 PAINT 108,120,2
70 LINE 0,100,200,98,1
80 PAINT 10,140,3
90 TEXT 10,10,"WIND   SURFING",1,3,8
100 PAUSE 7
```

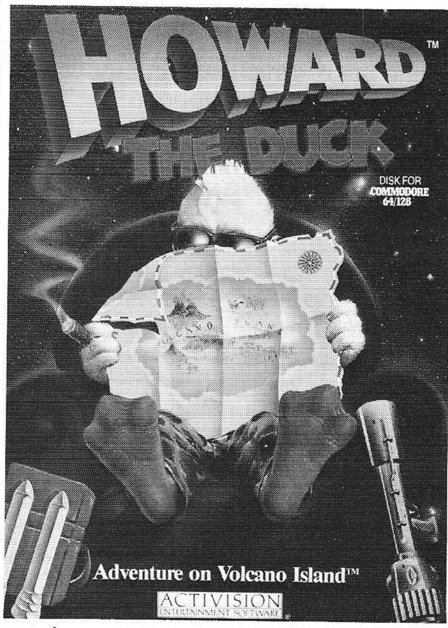
```
10 HIRES 1,1:MULTI 6,2,4
20  A=INT(75*RND(1))+1
30 B=INT(75*RND(1))-1
40 C=INT(5*RND(1))
50 CIRCLE 80,100,A,B,C
60 GOTO 20
```

Van het pakket met inzendingen houd ik nog wat over. Enkele heb ik terzijde moeten leggen. Dit zijn de klachten: ze lopen niet, ze zijn met de hand geschreven en niet te lezen, er staat geen afzender bij of het wemelt van de grafische tekens, die mijn tekstverwerker niet kent.

Nico Baaijens

Howard the duck

Je twee beste vrienden zijn verdwenen en worden gevangen gehouden in een actieve vulkaan. De speler, in dit geval een eend, moet ze uit deze benarde positie zien te bevrijden. Daartoe komt hij aanvliegen met een parachute op het eiland waar de vulkaan staat. Dan is hij er echter nog niet, integendeel!
Allereerst zijn er een heleboel hindernissen die onze eend moet nemen. Ten eerste is er het mutanten-slijm, waar hij overheen moet zien te komen om bij zijn rugzak met uitrusting te komen. Vervolgens zijn er nog de mutanten zelf, die de eend ernstig kunnen kwetsen. Bovendien kan Howard, als eend zijnde, totaal niet over bomen en struiken springen. Dat maakt van de eilanden een waar labyrinth. En de tijd blijft doorlopen: na een half uur is het afgelopen en barst de vulkaan uit.
Het spel oogt erg grappig en het beeldscherm is full-colour. Verder is het spelidee goed uitgewerkt: het uitschudden van de staart, de aparte



manier van lopen en het waterskiën met straalaandrijving (?) zijn aardige voorbeelden. Het spel is op meerdere levels te spelen. Activision was de maker, Ariolasoft de distributeur. Op tape en cassette.

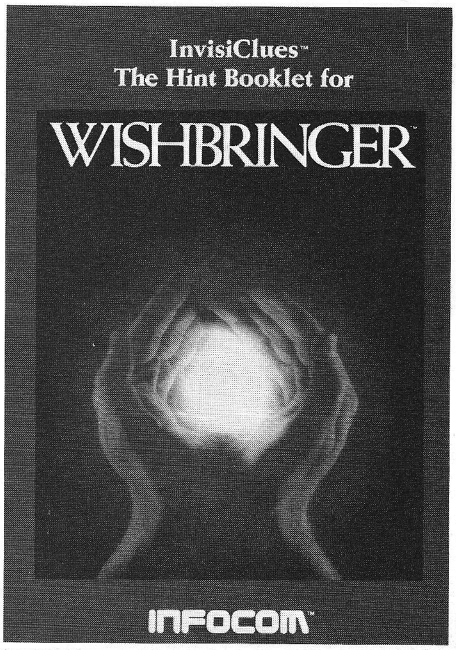
Wishbringer

Voor de (verborgen) avonturiers onder u die vinden dat het alledaagse leven wat al te saai is, zal een adventure wellicht een goede mogelijkheid zijn eens lekker risicoloos uit de bol te gaan. Infocom's laatste telg **Wishbringer** schept deze ambiance op een uitstekende manier.
In Wishbringer moet men in de rol van postbode een geheimzinnige brief aan de eigenaresse van een magiewinkeltje bezorgen. Na zessen, wanneer men de brief afgeleverd heeft, blijkt dat het gemoedelijke stadje Festeron, waar het verhaal speelt, is omgevormd tot het enge Witchville. De brug naar de stad wordt door trollen bewaakt en ook uit de lucht wordt u bespied. Het draait allemaal om een geheimzinnige steen, die de eigenaar in staat stelt allerlei wensen te doen (die ook nog uitkomen). Deze steen is de speler op slinkse wijze door de eigenaresse van de magiewinkel in de maag gesplitst. Aan u dus de taak om 'the evil one' (de slechterik in het verhaal) te overmeesteren en de kat van het oude vrouwtje, die door hem ontvoerd is, terug te brengen.
Bij de diskette voor C-64/128 (in fraaie verpakking) wordt een eenvoudige kaart, een plastic steen en een uitgebreide handleiding geleverd. Deze handleiding, weliswaar in het Engels, bevat een duidelijk overzicht van alle te gebruiken comando's. Wishbringer spreekt overigens een aardig mondje engels; men kan gebruik maken van echte volzinnen en ook afkortingen zoals 'it', die het tikwerk van de avonturier veel makkelijker maken, worden begrepen. Naast de woordenlijst is ook de complete legende van Wishbringer opgenomen alsmede een aantal handige tips voor beginners. (maak een kaart, save de spelsituatie vaak, etc.)
Zelfs de meest gehaaide avonturier loopt wel eens vast. Daarom is er het (optionele) hintpakket, bestaande uit een Hintboek en een zeer uitgebreide kaart. Het Hintboek is een van de originele aspecten van Wishbringer: men kan de antwoorden van belangrijke vragen (bv. "wat moet ik doen als ik bij de poort kom?") zichtbaar maken door over het schijnbaar lege hokje te strijken met een bijgeleverde pen. Zo wordt het spelplezier van de avonturier niet bedorven als hij het antwoord op een bepaalde vraag niet weet, maar de rest liever zelf wil uitvissen.
Infocom is een softwarehuis dat zich voornamelijk bezighoudt met het ontwerpen van adventures. Haar zegetocht begon in 1983 met het eerste deel van **Zork**, nu een klassieker voor de C-64. Zork was een van de eerste tekstadventures voor de homecomputer. In tekstadventures wordt er in het verhaal geen gebruik gemaakt van graphics (plaatjes). De grote kracht en spanning haalt de adventure uit de interactie. Met interactie wordt bedoeld dat iedere handeling van de speler zijn invloed heeft op het verdere verloop van het spel. Een oude man langs de weg van een dolende avonturer geeft doorgaans meer inlichtingen als men hem bijv. vooraf een biertje aangeboden heeft.
In een goede interactieve adventure is

Star soldier

Star soldier is een echte wargame, en dan zo een in de trant van Rambo I en II, Commando, etc. Het gaat er dan nu ook weer om zoveel mogelijk slechterikken neer te knallen ten bate van jezelf. Bij Star soldier ligt het speldoel toch iets anders: de speler verplaatst zich in de rol van huursoldaat. Voor iedere opgeknapte klus krijgt hij een hoeveelheid geld, waarvan hij zich weer van wapens voorziet. Er kan echter ook energie gekocht worden, een rang in het huursoldatenleger is alleen te koop, en zelfs voor het afbreken en saven van het spel moet geld betaald worden. Om zich van extra liquiditeiten te voorzien kan men natuurlijk ook op rooftocht uitgaan.
Men kan met een soort focus allerlei planeten in ogenschouw nemen, kijken wat wat voor klus er opgeknapt moet worden, wat de betaling is en andere bijzonderheden van de planeet, zoals het klimaat. Ook de aarde is in het overzicht opgenomen!
De graphics, en daar gaat het toch om (Star Soldier is een echt doe-spel), zijn alleszins redelijk te noemen als men ze met die van programma's als Commando vergelijkt. De muziek die onderwijl gespeeld wordt is goed en houdt de spanning er echt in. Al met al is deze nieuwste wargame van Quicksilvia een redelijk gelukt exemplaar. Op tape en diskette.

het vaak mogelijk op meerdere manieren het beoogde doel te bereiken. Dit is bij Wishbringer ook het geval. Een geslaagde adventure, zoal naar onze mening Wishbringer is, moet de creativiteit van de speler de ruimte geven.



En Infocom is daar tot nu toe erg goed in geslaagd.
Bij Wishbringer ontbreekt het de speler in geen geval aan interactie en mogelijkheden. Wij vonden het zelfs beter dan **Zork III** (The Dungeon Master). Een groot nadeel van dit programma is natuurlijk weer de taalbarriere. Wil men de adventure redelijk spelen, dan moet men over een behoorlijke kennis van de Engelse taal beschikken. Bovendien is het programma alleen op disk verkrijgbaar. Tekstadventures beslaan, mede door de vele mogelijkheden, veel meer geheugen dan de 57K vrij programmeerbaar in de C-64. Daarom zal vaak steeds opnieuw een stukje van het spel ingeladen moeten worden. Een diskdrive is dus noodzakelijk. Voor de mensen die aan bovenstaande voorwaarden voldoen, is deze nieuwste topper van Infocom, naar onze mening een 'must'.

Articfox

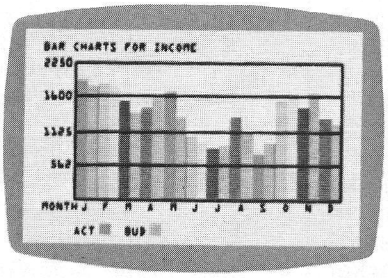
Van Electronic Arts is dit simulatiespel met de futuristische Articfox tank (gevechts-machine?). Als bestuurder heb je zeer vele mogelijkheden om je vijan-

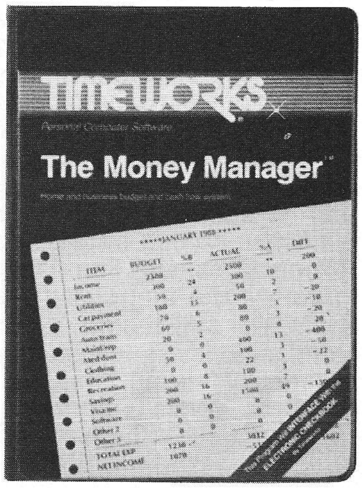


den te lijf te gaan, de meeste bestaan uit de langzamerhand bekende lasers en geleide projectielen. Dit "out-of-the-window" spel is grafisch zeer mooi en Aackosoft heeft het dan ook weer in de promotie gegooid voor *f* 59,90 op disk. De moeite waard, al is het genre langzamerhand overbekend. ●

Het Amerikaanse softwarehuis Timeworks Inc. heeft zich toegelegd op de serieuze en (semi)-zakelijke programmatuur voor de C-64 en C-128. Haar tekstverwerkers, spreadsheets, datamanagers, desktop- accessoires en financiële planners maken uw Commodore tot een echte zakelijke partner. Timeworks laat hiermee weer eens zien dat de serieuze software niet alleen is voorbehouden aan de MS-DOS, 68000- en mini-computers.

Timeworks

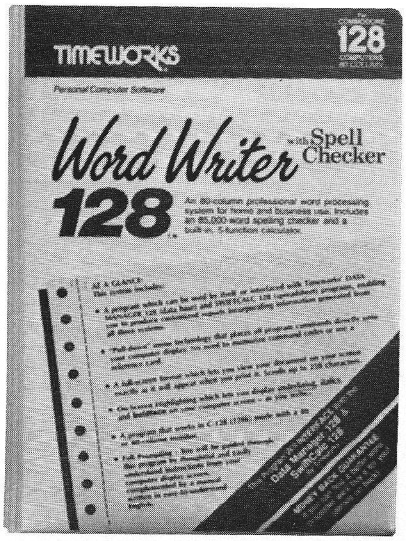




Serieuze software die op tijd werkt

Timeworks heeft op de Amerikaanse markt een goede naam als leverancier van Home Management-, Personal Improvement-, Business Systems en Education software opgebouwd. Het softwarehuis was een van de eerste, dat zich realiseerde dat het de C-64 (en later ook de C-128)-bezitters niet alleen om de spelletjes te doen is. Overigens geen kwaad woord over de huidige kwalitatief hoogstaande Commodore videogames, maar de ware hobbyist wil zijn of haar computer ook voor het zwaardere werk gebruiken.

Het ontwerpen van een paar kwalitatief goede tekstverwerkers, educatieve software, financiële planners, spreadsheets, en databases alleen is natuurlijk niet genoeg voor het veroveren van een sterke marktpositie. Daar komt meer bij kijken. Volgens Timeworks is customer support (klantenondersteuning) de formule tot succes. Elke geregistreerde gebruiker heeft recht op antwoord omtrent problemen of vragen. In de VS werd daartoe een hulpteam opgezet dat gratis (toll-free) op de helplijn gebeld kan worden. Een andere attractieve vorm van service verlening is de vrije upgrade en exchange politiek.



Sinds kort heeft Timeworks haar terrein ook naar Europa verlegd en wordt de software ook in ons land verkocht, o.a. door Cafka. Een reden voor Commodore-Info om deze veelbelovende serieuze software eens te bekijken.

Tekstverwerking

Voor de C-128 biedt Timeworks de krachtige word processor Word Writer 128. Deze tekstverwerker biedt de gebruiker alle moderne faciliteiten zoals verplaatsen/ kopiëren, zoek en vervang, invoegen, wissen, formatteren, automatische word wrap, tabuleren, inspringen, standaardbrieven, highlighting (ook voor onderstrepen,pen vet- en schuinschrift), kop/ voetnoten, super- en subscript, automatische paginanummering, het koppelen van documenten, speed keys, index en horizontaal/vertikaal scrollen.
Natuurlijk is deze tekstverwerker professioneel en dus op 80-koloms. Trekmenu's vergemakkelijken het geven van opdrachten en het full screen

formaat laat alle hoeken en gaten van het tekstfile zien.
Extra opties zijn de Spellings Checker en een Thesaurus. Leuk voor Engelse werkstukjes, maar minder handig in het Nederlandse taalgebied. Verder is Word Writer geheel compatibel met de Sylvia Porter-reeks, Datamanager 128 en SwiftCalc 128. De prijs ligt rond de 150 gulden.
De 40-koloms versie van Word Writer is vooral voor de C-64 bedoeld, maar draait uiteraard ook op de C-128 in de 64-modus. Dit pakket werkt met gewone menu's en een toetsenbord-overlay voor een gebruiksvriendelijke bediening op de C-64. Als extra's zijn aanwezig een spellingschecker (van 85.000 woorden), en een thesaurus met vele duizenden Engelse synoniemen en een calculator met vijf rekenfuncties. Ook kan de gebruiker in combinatie met Datamanager of SwiftCalc mail-merge persoonlijke standaardbrieven verzenden en/of etiketten afdrukken.
De C-64 versie van Word Writer kost rond de *f* 150,-, o.a. bij Cafka.

Databases

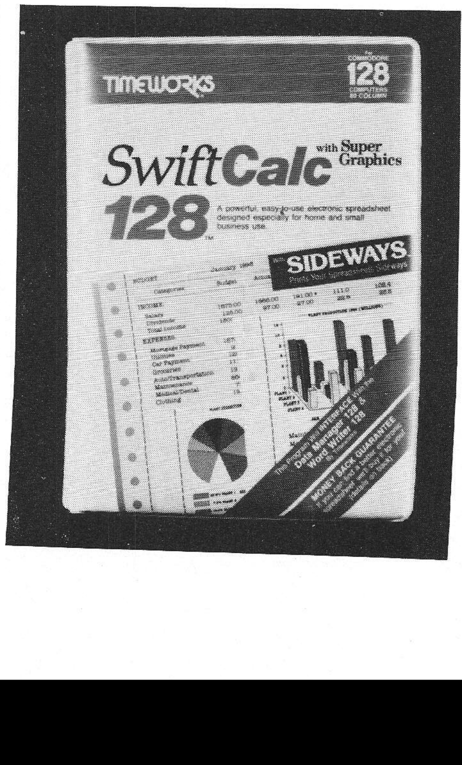
Ook van de DataManager zijn er weer een aparte C-128 en C-64 uitvoering verkrijgbaar. De kracht van de C-128-versie is de presentatie in 80-kolommen op een daarvoor geschikte monitor. DataManager 128 is volledig trekmenu gestuurd en een database met spreadsheetachtige mogelijkheden. Krachtige opties zijn X-SEARCH (werkt als een soort kruistabel) en X-SORT (sorteert alfabetisch, numeriek en chronologisch op datum. Middels het doorrekenen van kolom naar kolom en veld naar veld laten de standaardtoepassingen zoals betalingen van salarissen, kostenramingen e.d. zich eenvoudig realiseren.
Er zijn twee soorten graphics: de X-CHART en de STANDARD BAR CHARTS. Daarmee kan de gebruiker de cijfers over maximaal 10 index-categorieën gemakkelijk visualiseren. De statistische analyse omvat de opties SUM, AVERAGE en STANDARD DEVIATION. Niet veel, maar toch voldoende voor de doorsnee Commodore-bezitter.
De printopties bieden o.a. rapporten op maat, standaard postlabels en het verwerken van teksten in etiketten. Verder kunnen de worksheets direkt in de Word Writer tekstbestanden opgenomen worden. DataManager 128 kost circa 150 gulden.
DataManager 2 tekent slechts 40-kolommen op het scherm, maar draait daarentegen zowel op de C-128 als de C-64. Er zijn geen trekmenu's. De overige opties zijn min of meer gelijk aan DataManager 128. DataManager 2 kost circa 100 gulden.

Spreadsheets

SwiftCalc 128 en **SwiftCalc (64)** zijn Timeworks spreadsheets voor de Commodore computers. Beide pakketten kunnen als stand alones of in combinatie (interfacing) met de DataManager en Word Writer software gebruikt worden.
De spreadsheets zelf beschikken over de volgende functies:

- 80-koloms.
- Quick Entry. Automatische insert van kolom- en rijopschriften voor dagen, maanden en sequentiële nummering.
- Search & Replace. Opzoeken en automatisch vervangen van data.
- Insert/Delete van kolommen met een toetsaanslag.
- Formatting als Decimaal (9 cijfers), Graphics, Exponentiële notatie, $-teken en komma's.
- Automatische cursorbeweging in de door de gebruiker gespecificeerde richting.
- Move/Copy van de celinhoud.
- Insert/Delete van rijen en kolommem
- Swiftkey voor het benoemen van hotkeys voor uitgebreide opdrachten.
- Swiftsort voor het rangschikken in op/aflopende volgorde.
- Browse Mode voor het "doorbladeren" van het werkblad.
- "Freeze" voor het immobiliseren van de kolommen bij werkbladbewerkingen.
- Disk Cataloging voor het katalogiseren en indexen van de werkdiskette.
- "Locking" Ability beschermt tegen per ongeluk uitwissen van de celdata. Beschermt ook tegen overwriting.
- SwiftDOS voor een uitstapje naar het Commodore Diskette Operating System terwijl de spreadsheet in RAM blijft.
- 250 rijen bij 250 kolommen, in totaal 62.500 cellen.
- De mathematische functies minimum, maximum, gemiddelden, integere/absolute waarden, exponenten, sinus, cosinus, tangens, arctangens, logaritmen (log en ln) en de natuurlijke exponent e.
- De financiële analyse functies voor de huidige en toekomstige waarde van hypotheken, schuldbedragen of munt-eenheden.
- Help-schermen.
- Diverse educatieve voorbeeldwerkbladen en -modellen.
- Het hulpprogramma SIDEWAYS voort het vertikaal, roteert de spreadsheetkolommen over 90 graden, afdrukken van het werkblad.
- Het hulpprogramma SUPER GRAPHICS voor het maken van pies, vertikale staafgrafieken, lijngrafieken, scattergrafieken en 3-D staggered bar charts.
- What...if-analyse.
- Interfacing met Word Writer- en DataMager software.

SwiftCalc 128 kost circa 150 gulden. SwiftCalc voor de C-64 is enigzins beperkter van opzet. 80- Kolommen, 250 rijen bij 104 kolommen (25.000 cellen) en minder grafische mogelijkheden kunnen voor sommige gebruikers een bezwaar vormen. Daartegenover staan een lagere prijs, ongeveer 100 gulden, een een-toets-reset en een handige keyboard overlay.



Sideways

Het hiervoor genoemde programma Sideways is ook los verkrijgbaar. Sideways draait spreadsheet- en tekstbestanden 90 graden rondom hun as en drukt deze tot 65.000 tekens lang af. Kettingpapier is daarbij vanzelfsprekend een must.
Sideways werkt zowel met de C-64 als de C-128 en is compatibel met de Better Working Spreadsheet, Calc Now, Cal- Kit, Creative Calc, MS Multiplan, Practicalc 64, Practicalc II, Syncalc, Trio en ASCII-bestanden. Geschikte printers zijn o.a. de Commodore MPS-, Epson-, Okidata-, NEC-, Panasonic-, Star- en Seikosha 550/1000-modellen.
De C-64/128 (40-koloms) en C-128 (80-koloms) uitvoering kosten elk rond de 60 gulden.

Partner 128

Het desktop-pakket van Timeworks is Partner 128. Deze cartridge biedt de gebruiker de volgende bureauhulpjes in electronische vorm.

- Een kalender/agenda. Met één toetsindruk zijn alle afspraken voor een bepaalde datum oproepbaar. Ook kunnen de afspraken per week of maand, naar keuze in agenda- of kalenderformaat, bekeken worden. Vaste afspraken zijn voorzien van een opvallende markering.
- Notitieblokje.
- Namen- en adressenlijst.
- Een calculator met printeroptie.
- Schrijfmachine, de printer typt elke toetsaanslag direkt uit.
- Adressenprinter voor enveloppen en labels.
- Screen Print voor het maken van hardkopies van het beeldscherm.
- Telefoonklapper met auto dial. (Deze optie werkt alleen met een auto dial-modem).
- SwiftDOS geeft op elk moment, ook met een ander programma in RAM, toegang tot de Commodore DOS-commando's,
- SwiftLock beveiliging middels een geheime keyboardcode.
- SwiftLoad. Een speed-up utility voor het laden op de 1541.
- Extended printer control voor het softwarematig instellen van de verschillende schriftsoorten.

Partner 128 kost in de C64/128 (64K, 40-koloms) versie 130 en in de C128 (128K, 80-koloms) versie 150 gulden.

De financiële hulpjes

In de VS schijnt het vrouwelijke financieel genie Sylvia Porter furore te maken. Deze Sylvia Porter is de hoofdredactrice van het gelijknamige Personal Finance Magazine, in Amerika een toonaangevend blad voor wie zuinig met zijn geld wil omgaan. Sylvia Porter heeft inmiddels twee financiële pakketten voor de Commodore 128 en 64 onder het Timeworks-label uitgebracht. natuurlijk in het Engels, maar ook hier gesignaleerd.

Financial Planner

Dit is een echt home managementpakket voor de eigen uitgaven. Geen simpel huishoudboekje, maar een doordacht financieel pakket dat behalve helpt regelen ook een stukje onderwijs in geldzaken biedt. In Financial planning by objectives



leert de gebruiker stapsgewijs hoe hij/zij een financiële planning voor carrière, de huwelijksstatus, de kinderen, verzekering, sparen, een bepaalde levensstijl, kleine investeringen, pensioen en mogelijke tegenslagen (o.a. ziekte of invaliditeit) op te zetten. Dit alles volgens een eenvoudig vraag- en antwoordsysteem.
Met **budget preparation** worden gedetailleerde maandelijkse vergelijkingen en jaarlijkse voorspellingen opgezet. Maximaal kunnen daarbij 140 verschillende items worden ingevoerd.
De **financial statement preparation** prepareert, houdt een oogje in het zeil en rapporteert de financiële status. Kortom het gedetailleerd bijhouden van de activa en passiva plus de in- en uitgaven.
Verder zijn er nog opties voor het beheer van de aandelen en belastinghulpjes. Deze laatste optie is in Nederland niet zo bruikbaar door het verschil met de Amerikaanse situatie. Verder is de Personal Finance Planner volledige compatibel met Word Writer en de andere Sylvia Porter pakketten. Het is in een 40-koloms en 80-koloms (allen de C-128) uitvoering verkrijgbaar. De prijzen liggen respectievelijk rond de 125 en 175 gulden.

Investment Manager 128

Dit is een zogenaamde portfolio manager. Het programma helpt de gebruiker dus bij het beheer van de aandelenportefeuilles. Inmiddels zijn er al diverse beurspakketten voor de Commodore-machines in omloop, maar Sylvia Porter's Investment Manager is zeker een van de veelzijdigste.
Behalve de gebruikelijke opties zoals het bijhouden van de gedane transacties over maximaal 15 verschillende portfolio's (aandelenpakketten); automatisch updaten van veranderingen, het bijhouden van de posities, winsten/verliezen en prijzen; het waarschuwen voor investerings deadlines (wanneer moet ik kopen/verkopen); het creëren van een oude-dags-voorziening en het beperken van de belastingaanslag biedt dit pakket nog een aantal extra's. Via een modem is er toegang tot bestaande 300/1200 baud beurslijnen en databanken (dus niet via Viditel). Met behulp van de fundamental & technical analysers is het mogelijk om de beursontwikkelingen grondig in de peiling te houden. Zelfs met behulp van verklarende grafieken voor prijzen, waarden, ratio's, percentages e.d. in kleurige staven en taartpunten.

Timeworks gaat met haar tijd mee. Zowel de serieuze C-64 en C-128-bezitters zullen bij deze zakelijke software veel van hun gading vinden. Een nadeel blijft de Amerikaanse opzet. Weliswaar zijn de Engelse versies al aardig aan de Europese markt aangepast, maar enkele opties zijn niet helemaal compatibel met de Nederlandse situatie. Dat doet overigens niets af aan de kwaliteit. ●

Met de zogenaamde "Braillecom 1" introduceert THL Computers, op basis van een C-64, een systeem, waarmee de communicatie tussen blinden en zienden wordt opengebroken. Daarmee wordt het voor elke visueel gehandicapte mogelijk geheel zelfstandig in het zogeheten zwartschrift teksten te schrijven.



Braille

De communicatie tussen twee werelden

Met de Braillecom 1, die de brailleteksten vertaalt in zwartschrift, zullen blinden in de toekomst veel zelfstandiger in de samenleving kunnen functioneren, waardoor hun kansen en mogelijkheden aanzienlijk worden vergroot.

Het systeem, een puur Nederlandse vinding, kost *f* 3.265,- exclusief BTW en ligt daarmee binnen het bereik van - in principe - elke blinde. Bovendien hebben de Gemeenschappelijk Medische Diensten (GMD) met belangstelling gereageerd en heeft men al in een aantal gevallen toegezegd de Braillecom 1 in bruikleen te verstrekken.
Zowel in de de onderwijswereld als voor het gebruik in de privé-sfeer biedt deze nieuwe communicatievorm geweldige mogelijkheden. Zeker, wanneer men bedenkt dat het braille-vertaalprogramma nu al beschikbaar is in het Nederlands, Duits, Frans en Engels.

Perkins Brailler blijft

Het systeem werkt in principe zeer eenvoudig. Bij de Braillecom 1 blijft de blinde gewoon gebruik maken van zijn Braille-schrijfmachine, de zogeheten Perkins Brailler. Deze typemachine, met daaronder een printplaat, is verbonden met de Commodore 64. Met behulp van het braillevertaalprogramma, zet de computer de ingetikte brailletekens direct om in zwartschrift. Deze tekst, die zichtbaar is op het beeldscherm, kan vervolgens worden opgeslagen op floppy, maar kan ook worden geprint. De Braillecom 1 vertaalt niet alleen leestekens, maar ook wiskunde, scheikundige formules en natuurkundige grootheden.
Bovendien is het systeem zodanig uitgevoerd, dat het eenvoudig mogelijk is om de Commodore 64 ook nog te gebruiken voor de bekende toepassingen zoals tekstverwerking, administratie, educatie en ontspanning.

Schoolgebruik

Op school verruimt de Braillecom 1 de communicatie tussen de blinde en diens docent. Onder meer is hij hiermee in staat huiswerk, tentamens en proefwerken aan te leveren, die ook voor de leraar direkt leesbaar zijn. Deze documenten behoeven nu niet langer te worden opgestuurd naar een vertaalbureau hetgeen geld en kostbare tijd bespaart.
In de privésfeer is de blinde vanaf nu in staat brieven te schrijven aan zienden, boodschappenlijstjes te maken, berichten voor gezinsleden achter te laten etc. Bovendien kunnen familieleden de blinde nu eenvoudiger ondersteunen bij het maken van huiswerk e.d.

Toekomstplannen

Naast de Braillecom 1 wordt er nu ook reeds gewerkt aan nieuwe uitbreidingen van dit systeem. Binnen afzienbare tijd zal **THL Computers** haar produkt-lijn voor blinden uitbreiden met een aantal nieuwe produkten, ondermeer met een systeem dat tevens het zwartschrift omzet in braille en een rekenmachine. Niemand zal dan ook raar opkijken als er binnen een paar jaar een veel groter scala van mogelijkheden is ontstaan. Zeker binnen het zakelijke circuit zijn legio mogelijkheden te bedenken, die de toegang van blinden tot het maatschappelijk verkeer vergroten. Er kan dan ook gesproken worden van een strukturele positieverbetering van blinden dankzij dit unieke vertaalsysteem.
Inl. 072-118504. ●

Onder de topmerken op spelgebied blijft Activision de meest hitgevoelige. Men gaat mee met de trends en uit de vooraankondigingen van nieuwe spellen voor de rest van dit jaar blijkt men veelal aansluiting te zoeken bij de trends op film en sportgebied.

Actief Activision

Het grote amerikaanse software-huis Activision legt zich al jaar en dag toe op het produceren van spellen, adventures en andere vrijetijdsprogramma's. Opvallend voor de nieuw gepresenteerde programma's van deze spellen-bouwer, is het nagenoeg ontbreken van oorlogsspellen. In de branche van tanks, bommenwerpers en ander oorlogstuig lijkt flink de klad te zitten, wat het beste doet hopen voor de wereld buiten de computer.

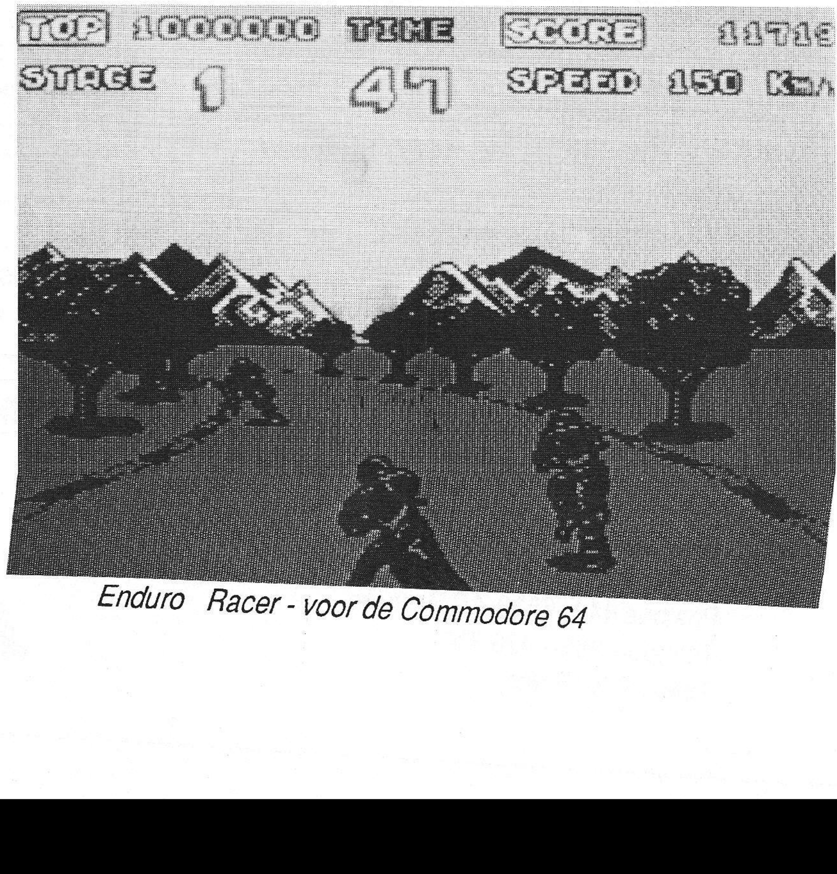
Het bedrijf werkt niet alleen voor de eigen software, maar heeft ook de licenties van een aantal andere software-huizen op zak, zoals Data East Atari Games, Bally Sente, en het Japanse Nichibutsu.

Bureaucracy

Voor iedereen die nog nooit in aanraking is geweest met de ware bureaucratie, of voor iedereen, die de eigen frustraties over de ambtelijke papierwinkel van zich af wil gooien is er nu het dolkomische Bureaucracy.
Een spel dat begint in je gloednieuwe appartement (in hartje New York) waar je net rustig een krantje wilt gaan lezen, als je baas opbelt om je te vertellen dat je naar Parijs moet. Nog steeds niets aan de hand zul je denken, dus op reis. Maar dan begint de ellende: het geld dat je baas overmaakt voor je reis komt niet aan, dus je moet zelf uitzoeken wat er mis is gegaan bij de PTT. En binnen de kortste keren ben je verwikkeld in een reeks van bureaucratische ongelukjes, traagdraaiende molens en spoorloos zoekgeraakte documenten. Als je dan op weg naar Parijs ook nog te maken krijgt met een wel erg strenge bank, een vliegveld waar iedereen stapelgek schijnt te zijn geworden en een paranoide eigenaar van een huisje in de bush-bush staat het huilen je nader dan het lachen. Als je door je eigen onoplettendheid dan uiteindelijk jezelf terugvindt, ondersteboven aan een boom bengelend in de jungle, terwijl je op je horloge de minuten ziet wegtikken voor die belangrijke afspraak in Parijs, dan weet je dat je de uiterste grenzen van de burocratie - en waarschijnlijk ook van jezelf - bereikt hebt... Kortom: een prima spel voor elke zich 'Yup' wanende avonturier, ambtenaar en minister van Sociale Zaken (vanaf de zomer leverbaar voor de Commodore 128).

Firetrap

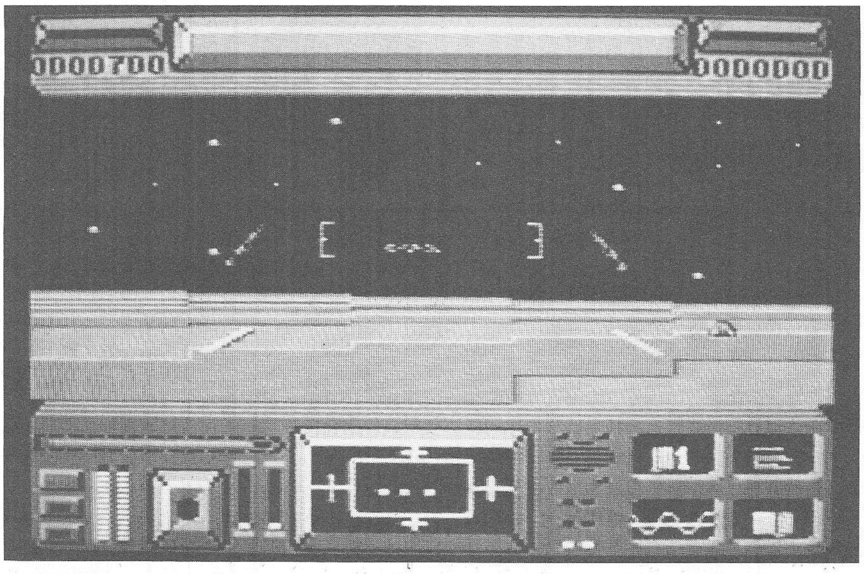
Als je eigenlijk altijd hebt gedroomd van een carrière bij de vrijwillige brandweer biedt Firetrap een goede mogelijkheid tot zelftraining. Gewapend met waterbommen, twee goede handen en een enkel ander hulpmiddel heb jij de eer om mensen te redden, die opgesloten zitten in een brandende wolkenkrabber. Vallende stenen, in brand vliegende auto's en andere, soms wat onduidelijke, obstakels maken het geheel nog een beetje lastiger. Na afloop een koel pilsje of een glas cola met ijs is wel het minste wat je mag wensen (binnenkort in ons land voor alle modellen Commodore).



Enduro Racer - voor de Commodore 64

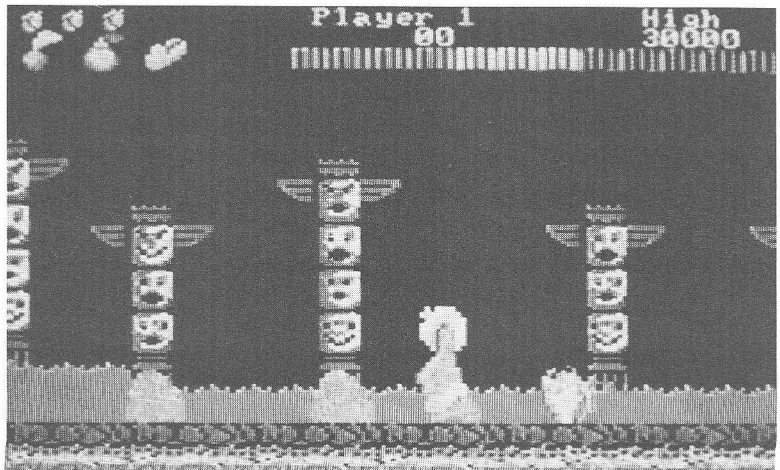
Supersprint
Custom Cars zijn aan de andere kant van de grote plas al jaren een ware rage. Hele volksstammen gaan in het weekeind het platteland op om de zelfgebouwde bolides te bewonderen en te genieten van sensationele races. Hoewel het er in je eigen (huis- of zolder-) kamer niet zo spectaculair aan toe gaat als op de dorre grasvlakten van Ohio, kun je als speler toch heel wat plezier beleven aan dit pakket. Gedurende een aantal manches moet je proberen zoveel mogelijk punten bij elkaar te sprokkelen. Natuurlijk is het parcours bezaaid met verborgen afsteekroutes, plotselinge obstakels, olieplassen en botsingen tussen concurrenten (tenzij je pech krijgt). De gouden stukken gereedschap, die per ronde gewonnen kunnen worden zijn essentieel in het hele verloop van de race. Daarmee kan namelijk extra turbo-power, grotere acceleratie of een hogere snelheid aangeschaft worden. Op die manier zijn er wel 15 verschillende bonussen te besteden gedurende een spel. Maar pas wanneer de andere tegenstanders verslagen zijn mag je met de champagne in het rond spuiten en je naam toevoegen aan de winnaars van de Supersprint-trofee (verkrijgbaar na de zomer).

Enduro Racer
De kwaliteit van dit cyclecross-spel is behoorlijk hoog. Met je eigen fietsje stuiter je over het parcours, waarbij gigantisch hoge snelheden haalbaar zijn. De gevolgen van een val zijn dan ook navenant. Maar al met al is het wel opletten geblazen. Stoffige zandsporen, blubber en hoge hellingen, rotsblokken, waterpartijen en tegenstanders: het is maar een deel van de hindernissen die genomen moeten worden. Een leuk spel, dat inspeelt op de fiets-cross rage. Maar als ik heel eerlijk ben: op volle snelheid de macht over je stuur verliezen levert zo'n fantastische crash op dat ik dát moment eigenlijk het fraaiste vindt (waarschijnlijk al vanaf april in ons land verkrijgbaar).



Star Raiders

Wonderboy
"Zoals elke jongen er van droomt om een Wonderboy te zijn in de ogen van zijn grote liefde, zo droomt elk meisje van de jongen van haar hart, die er alles voor over heeft om haar te bereiken." Dat is waarschijnlijk de basisfilosofie geweest van de makers achter dit spel, dat je als held-op-sokken langs bergen en dalen, slangen en rollende keien naar het meisje-met-de-blonde-lokken voert.
Hoewel onvoorstelbaar, kun je in dit zeer onvriendelijke landschap soms toch gebruik maken van een skateboard(!), een bijl en andere nuttige gereedschappen om je grote liefde uiteindelijk in je armen te kunnen sluiten (vanaf juni voor alle C 64 en 128 machines).



Wonderboy

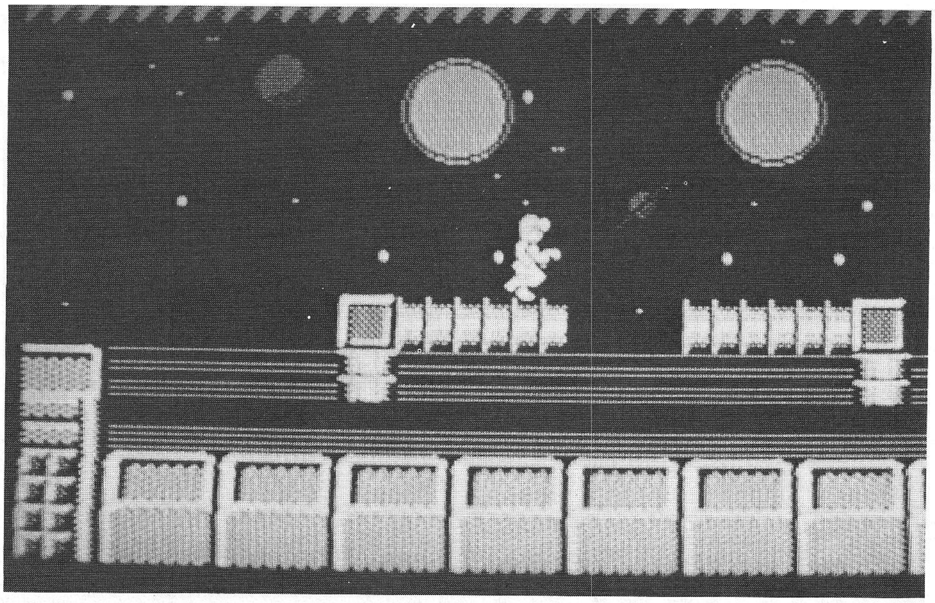
The Quartet
Een spel waarin de grafische kwaliteiten van de huidige machines volledig worden uitgebuit is The Quartet. Weliswaar het overbekende thema van de strijd in en om de ruimte, maar wie daar doorheen kijkt heeft er veel plezier aan. Drie mannen en een vrouw moeten een ruimte-kolonie bevrijden van de 'Baddies'. Via een reeks van opdrachten komt men steeds een niveau verder tot men uiteindelijk de Grote Boef zelf te pakken heeft. Veel schieten met laserwapens, vliegen met speciale pakken aan en 'jet'-laarzen om reuzesprongen te maken. Twee spelers kunnen onafhankelijk van elkaar opereren, wat het spel een extra dimensie geeft (vanaf mei te koop in de betere computershops).

Rampage
In Amerikaanse speelhallen zou je dit spel al wel eens aangetroffen kunnen hebben. Rampage is een spel voor één tot drie spelers, die als Lizzy, Ralph of George veranderen in enorme nietsontziende film-monsters. De enige manier voor dit anders zo vriendelijke drietal om daar weer vanaf te komen bestaat uit een reeks van opdrachten, vechtpartijen en veel be-

hendigheid. Omvallende huizenblokken, wolkenkrabbers, helicopters, het hele leger achter je aan... Het klinkt vreselijk, maar dat is het allerminst. Het is juist een heel leuk en vermakelijk spel, dat beslist door iedereen gespeeld kan worden. En mocht King Kong plotseling zijn grote hand door het raam steken, dan kun je hem beter een joystick geven... Misschien is het je sympatieke buurman wel...(voor alle C-64 en 128 computers verkrijgbaar).

Star Raiders II

Menig bezitter van een home-computer (en zijn of haar huisgenoten) heeft al een enorme historie achter zich als piloot in het ongekend succesvolle ruimte-avontuur Star Raiders I, zoals dat door Atari ooit in de speelhallen is gebracht. Later bleek ook de thuis-versie hoog te scoren op de spelletjes-markt. De opvolger van dit pakket mag dus bijna bij voorbaat als (kas)succes gezien worden. Toch is het voortborduren op een eerder succes niet altijd even geslaagd. Natuurlijk ook hier weer de (onoverwinlijke) tegenstander Chut, als slechte heerser van Zylon, die van plan is het hele Celos IV-sterrenstelsel te vernietigen. Gelukkig ben jij nog in het bezit van het supersnelle ruimte-vliegtuig Liberty Star, de 'snelste schutter van het wilde westen... (oh pardon, van de ruimte). En verder is het een kwestie van schieten en beschoten worden, behendig manoeuvrerend met je joystick.

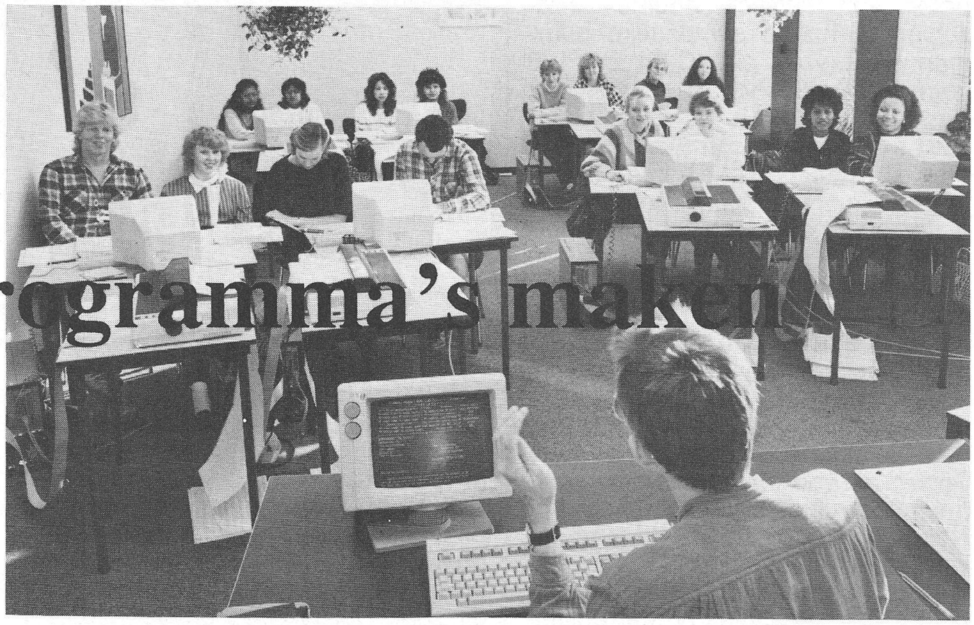


The Quartet

Ik val niet om van enthousiasme voor dit spel. Het is goed en de visuele vormgeving wel aardig. Maar in vergelijking met de andere hierboven beschreven spellen toch een stuk minder boeiend (op cassette en disk voor Commodore-bezitters, maar helaas pas aan het eind van het jaar) •

M. de Rooij.

In de vorige afleveringen van "Zelf leerprogramma's maken in Basic" ging het om specifieke leerdoelgerichte programmatuur. Centraal daarin stond het oefenen van bepaalde vaardigheden. U gaat nu kennis maken met een heel ander toepassingsgebied van de computer. Voor veel onderwijsgevenden waarschijnlijk geen bekend terrein. Toch zal blijken, dat een computer in de klas meer kan zijn dan alleen een leermachine. Bob Munniksma neemt u mee naar verre einders en avontuurlijke eilanden.



Zelf leerprogramma's maken in Basic

Deel 4: Op avontuur in de klas.

Dat een computer inderdaad meer kan zijn dan een leermachine, zullen veel onderwijsgevenden, die ervaring hebben met dit nieuwe medium, kunnen beamen. Denk aan het brede spectrum van mogelijkheden van de computertaal Logo. Binnen dat kader zijn veel wetenschappelijke onderzoeken gedaan en de conclusies zijn bekend. Niet de kinderen moeten door een machine (de computer) worden geprogrammeerd, maar de kinderen moeten de machine leren programmeren.

Van alle genres software voor de Commodore computers lijken de meeste gebruiksmogelijkheden voor het onderwijs te liggen bij de zogeheten *mind-games*, oftewel denkspelletjes.
Het spelen van adventures, ook wel avonturen genoemd, vereist een goed ontwikkeld denk- en associatievermogen. Het spelen van dergelijke mind-games scherpt die vaardigheden en ontwikkelt het probleemoplossend denken van de speler(ster).
De bestaande avonturen zijn echter *voor kinderen veel te moeilijk en complex*. Ervaren volwassen avonturiers zijn vaak maanden bezig met een pittig adventure zoals bijvoorbeeld The Hobbit. Niet geschikt voor kinderen dus. Ook zijn de meeste adventures in het Engels, dus ook die vallen voor leerdoeleinden af. Om een geschikt adventure te krijgen, zit er dus niets anders op, dan zelf aan de slag te gaan met het ontwikkelen van een adventure-achtig spel. En waarom eigenlijk niet. Hoe primitief de Basicversie van de C-64 ook is, er blijkt toch veel mogelijk.

Voorwaarden

Nu zal een dergelijk adventure voor kinderen aan een aantal voorwaarden moeten voldoen:

- °Het verhaal moet boeiend, uitnodigend en stimulerend zijn en blijven.
- °Taalgebruik en zinsstructuur moeten qua niveau zijn aangepast aan de doelgroep.
- °De structuur van het programma moet overzichtelijk zijn, zodat de onderwijsgevenden, die de adventures bedenken, niet te veel programmatische problemen ontmoeten.

Aan deze voorwaarden voldoen is geen geringe opgave. Het programma op deze en volgende bladzijden zal daarbij een leidraad zijn.

Het verhaal

Het verhaal dat hier is gekozen is simpel. Ergens op een eiland ligt een *schat begraven. De piratenkoning* Rooie Pier heeft deze schat ooit eens begraven. Om dat eiland te vinden en de schat op te graven, moet de speler een aantal gevaren trotseren en een

groot aantal beslissingen nemen. Het hele gebeuren is te vergelijken met een spannend boek. Met het verschil, dat in een boek de afloop vaststaat en in een adventure niet.
Wel liggen de mogelijkheden vast. Die worden bepaald door de programmatuur. Zo moeten de kinderen het verhaal ook zien: een spannend boek, waarin zij zelf de hoofdrol spelen. Het verhaal in ons voorbeeldprogramma is willekeurig. U kunt evengoed een verhaal uit een bestaand kinderboek als uitgangspunt nemen. Of neem een sprookje! Verder kunt u alle factoren zelf veranderen en beïnvloeden. Ik zal u vertellen hoe.

De structuur

Een avontuur heeft als grondslag een soort ingebouwde plattegrond met verschillende lokaties, waar al dan niet iets te zien is. De doorgang naar de andere lokaties is vastgelegd in de structuur van het avontuur. De avonturier moet aan bepaalde voorwaarden voldoen en bepaalde lokaties bezoeken. Beslissingen zijn van belang voor de afloop van het verhaal. Het verhaal begint in één van de lokaties en de afloop wordt geheel bepaald door de keuzes van de speler(ster). Uit het voorgaande is af te leiden, dat de programmatuur diverse secties zal hebben. Dat is inderdaad het geval. We zullen de listing eerst eens globaal bekijken. Daarna aandacht voor enkele details.

Initialisatie

In de regels 130-210 wordt het verhaal opgezet, de schermkleuren bepaald en variabelen gevuld. De startlokatie wordt willekeurig gegenereerd in regel 140 en het nummer wordt vastgelegd in variabele K.
Dan worden er een aantal array's ingelezen. Alle informatie is te vinden in de DATA-regels vanaf regel 1000. Het voordeel van een dergelijk aanpak is, dat aan deze DATA-regels eenvoudig een andere inhoud kan worden gegeven, bijvoorbeeld een geheel ander verhaal. Een alternatief zou zijn om alle lokaties onder te brengen in een subroutine en de informatie over de lokaties met eenvoudige PRINT-statements op het beeldscherm te zetten. Ik koos hier niet voor, omdat de structuur ondoorzichtiger zou worden en daardoor moeilijker aan te passen.
De betekenis van de informatie en de plaats in het programma komen nog aan de orde. Als alles is ingelezen, verschijnt de titel op het scherm met informatie over de start-lokatie.



ZORK is al jaren hét klassieke adventure-spel

De lokaties

Het programmadeel in de regels 815-925 bestuurt in feite het hele avontuur. Hier wordt gekeken naar welke lokatie de speler zich begeeft en als de gewenste aansluiting er inderdaad is, krijgt de speler de nodige informatie. Tevens wordt hier bijgehouden of er al aan de voorwaarden wordt voldaan om het avontuur succesvol af te sluiten. Ook de valstrikken worden hier uitgezet. Want een avontuur is pas spannend, als er een zeker risico bestaat.
Aangezien de start-lokatie altijd legaal is, springt het programma in het begin direct naar regel 825. In deze fase hebben de IF-statements van de regels 830-840 geen betekenis. Dat zal in een gevorderd stadium anders zijn!
De subroutine op 655 laat zien welke voorwerpen er op de lokatie aanwezig zijn. Deze voorwerpen hebben natuurlijk een belangrijke betekenis in het verhaal. U zult dat nog zien. Dan springt het programma naar een hele lange subroutine, die begint op regel 240.
In bepaalde situaties en op bepaalde lokaties wordt gekeken welke voorwerpen de speler bij zich heeft, resulterend in een voorwaardelijke programmasprong. U kunt dit programmadeel zelf verder uitbreiden met noodzakelijke voorwaarden voor de speler. Zo kunt u invloed uitoefenen op de moeilijkheidsgraad en de afloop van het verhaal. Zo kan de speler zonder boot niet naar het eiland. Ook een te groot aantal meegenomen voorwerpen verhindert de reis. Zie 820 en 860 (geen boot) en 885-900 (te zwaar).

Kunstmatige intelligentie

Nu is het van belang, dat het lijkt, alsof het programma reageert op de mededelingen van de speler. Een computer is een dood ding en kan dat natuurlijk nooit doen. Wel kunt u de regels zo programmeren, dat het er op lijkt. Er is een schijnbare interactie tussen mens en machine. Hoe volmaakter die interactie is, des te interessanter wordt het spel. In de subroutine, beginnend op regel 240, wordt een zekere mate van kunstmatige intelligentie gerealiseerd.
Het vocabulair is wel beperkt, maar binnen die beperkingen verloopt de conversatie heel aardig. U kunt de woordenschat van de machine zelf uitbreiden binnen de structuur van het programma.

Subroutine

Het verloop van de subroutine is als volgt: eerder gebruikte variabelen worden geleegd voor hernieuwd

gebruik. Dan vraagt de computer: WAT NU:? Er wordt een INPUT met een hekje gebruikt. Daarmee vervalt het vraagteken en alleen de cursor knippert achter de indringende vraag. Ja, inderdaad, wat nu. Hiermee begint het avontuur. Het antwoord wordt na invoer en een tik op <return>, verder aan een ontledend onderzoek onderworpen. Als het uit twee delen bestaat, gescheiden door een spatie, dan wordt het eerste deel in variabele A1$ gezet en het tweede deel in A2$. De spatie zelf doet niet mee. Zie de regels 255-280. Een enkelvoudige opdracht komt in variable A1$. Vanaf dit moment kan er worden teruggesprongen naar het stuurgedeelte door een RETURN-statement. Wanneer en waar dat gebeurt hangt af van de ingetoetste opdracht.
Daarom zult u in de rest van de regels van deze subroutine vele malen het RETURN-statement tegenkomen. Slechts één ervan wordt gebruikt als uitgang.

De opdrachten

Als u de DATA-regels naleest, ziet u vanzelf de mogelijke opdrachten:
NOORD, ZUID, OOST, WEST. Deze bepalen de reisrichting;
MEE. Hiermee kunnen de kinderen bekijken, wat ze op dat moment bij zich hebben.
HELP. Bij vastlopen verkrijgt de speler met deze opdracht een hint.
KIJK. Hiermee kan er gekeken worden of er een voorwerp aanwezig is op de lokatie en zo ja, welk voorwerp.
Dit zijn de enkelvoudige opdrachten. Verder:
PAK. Een voorwerp kan worden gepakt met deze opdracht.
WEG. Het weggooien van een meegedragen voorwerp gaat met dit commando.

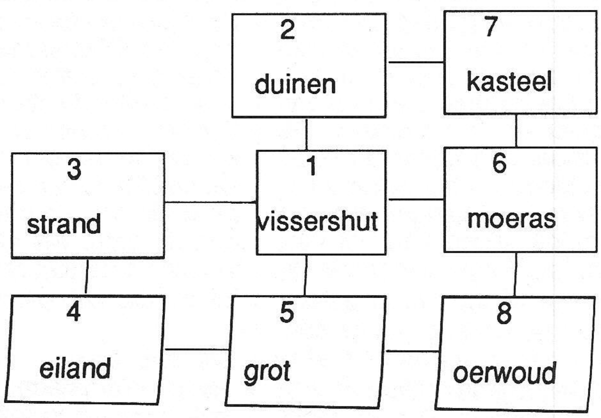
Bij het ingeven van de reisrichting kan ook worden volstaan met het intikken van een willekeurig aantal letters van het woord; voor NOORD voldoet N en NOO dus ook.
Als na KIJK ook nog de naam van een voorwerp wordt ingetikt en de speler heeft dat voorwerp inderdaad bij zich, verschijnt er enige informatie over dat voorwerp en vaak ook een aanwijzing.
Na PAK moet ook de naam van een voorwerp komen. Als het voorwerp op de bezochte lokatie aanwezig is, wordt het toegevoegd aan de invertaris. Bij WEG moet eveneens worden verteld, welk voorwerp weg moet. Steeds als opdrachten worden gegeven, die niet realiseerbaar zijn, maar wel tot de mogelijkheden behoren, zal het programma adequaat reageren, alsof het de speler wil corrigeren. Als de speler een niet aanwezig voorwerp wil pakken, wordt meegedeeld: *"Dat ligt hier niet."* Of bij het bekijken van een niet aanwezig attribuut: *"Je hebt geen ...... bij je."*
Als de opdracht geheel buiten het vocabulair valt, dan reageert de computer met: *"Ik begrijp je niet."*

Voorbeeld

Ik zal de analyse van een mogelijke opdracht met u doorlopen:
NOORD (of desgewenst: N)
De NOORD-opdracht zit in AN$(1) (zie DATA-leesregel 180). Door een lus te gebruiken, die de eerste vier elementen bekijkt en bij herkenning laat springen, bereikt u het gewenste resultaat. In dit voorbeeld blijft de variabele I dus 1 en er wordt gesprongen naar regel 375. Daar wordt

```
100 rem *******************************
105 rem **  avonturen voor kinderen   **
110 rem **  (c) 1987  bob munniksma   **
115 rem **     voor commodore info    **
120 rem *******************************
125 :
130 rem initialisatie
135 open 1,0:e=0:pb=0:at=0:sc=100
140 k=int(rnd(1)*7)+1:if k=4 then k=1
145 print chr$(147)chr$(14)chr$(8)
150 poke 53280,0:poke 53281,0:poke 646,7
155 for i=1 to 8:for j=1 to 4
160 read ii$(i,j):next j,i
165 for i=1 to 2:read i$(i):next i
170 for i=1 to 8:for j=1 to 3
175 read lk$(i,j):next j,i
180 for i=1 to 9:read an$(i):next i
185 for i=1 to 8:for j=1 to 4
190 read rl(i,j):next j,i
195 for i=1 to 8:read at$(i)
200 m$(i)=at$(i):next i
205 for i=1 to 8:read at(i):next i
210 for i=1 to 5:read h$(i):next i
215 :
220 rem introductie
225 print i$(1):print i$(2):print i$(1)
230 print:print:goto 825
235 :
240 rem antwoord analyse
245 print:an$="":a1$="":a2$=""
250 print"Wat nu: ";:input#1,an$
255 print:for i=1 to len(an$)
260 if mid$(an$,i,1)=chr$(32) then 270
265 next i:goto 280
270 a1$=left$(an$,i-1)
275 a2$=mid$(an$,i+1):goto 290
280 a1$=an$
285 :
290 for i=1 to 4
295 if an$=left$(an$(i),len(an$)) then 375
300 next i
305 if an$=an$(5) then 390
310 if an$=an$(6) then 425
315 if an$=an$(7) then 355
320 for i=7 to 9
325 if a1$=an$(i) then 345
330 next i:af=5:gosub 480
335 print"Ik begrijp je niet....."
340 print:return
345 on i-6 goto 445,500,550
350 :
```

afb.1 Plattegrond van het voorbeeld-adventure



de relatie-tabel van de lokaties bekeken. Als er een geldige verbinding is (het bewuste element van **array rl(k,i)** is ongelijk aan nul), dan krijgt variabele V de waarde 1 en RETURN dient als uitgang van de subroutine. Het controledeel (vanaf regel 815) neemt de taak over en de sprong naar de gewenste lokatie wordt gerealiseerd. Alles begint dan opnieuw, maar dan vanuit een andere lokatie.
Nog een voorbeeld: **PAK BOOT**
Zoals eerder gezegd komt PAK in variabele A1$ en BOOT komt in A2$. Daarna gaat de analyse verder. PAK is het achtste element van AN$(X). Er volgt deze sprongvolgorde: 290 tot en met 325 gaat zonder mankeren. Met waarde 8 in variabele I vervolgt het programma zijn loop op regel 345. Daarna gaat het programma verder op regel 500. (On I-6 GOTO...).
Als er geen tweede woord werd gegeven (het te pakken voorwerp) volgt een mededeling op regel 525, daarna een terugkeer naar het controledeel. Als er wel een woord werd gegeven, volgt de controle of dat voorwerp inderdaad in de bewuste lokatie ligt. Als dat niet zo is, wordt dat wederom gemeld en teruggesprongen via RETURN. Als het voorwerp er inderdaad ligt, wordt het aan de inventaris toegevoegd en eveneens teruggekeerd via RETURN.
Het WEG doen van een attribuut uit de inventaris heeft een identiek programmaverloop maar dan wordt het voorwerp uit de inventaris verwijderd.



```
355 rem kijk zonder voorwerp
360 if at(k)=0 then print"Niets bijzonders
    te zien.":af=2:gosub 480:return
365 print"Er ligt een ";at$(k):af=5:
    gosub 480:return
370 :
375 if rl(k,i)<>0 then v=1:return
380 v=0:gosub 795:return
385 :
390 rem inventaris
395 print:print"Je hebt bij je: ";
400 if at=0 then print"NIETS !":return
405 print:print:for a=1 to 8
410 print a;me$(a):next a
415 af=5:gosub 480:return
420 :
425 rem help
430 h=int(rnd(1)*5)+1:af=10:gosub 480
435 print:print h$(h):return
440 :
445 rem kijk naar voorwerp
450 print:for i=1 to 8:if a2$=me$(i)
    then af=2:gosub 480:goto 460
455 next i:af=5:gosub 480
460 if me$(i)<>a2$ then print"Je hebt
    geen "a2$" bij je.":print:return
465 for kk=1 to 4:print ii$(i,kk):next kk
470 return
475 :
480 rem score bijhouden
485 sc=sc-af:if sc<0 then sc=0:goto 735
490 return
495 :
500 rem pak
505 if a2$="" then 530
510 if a2$<>at$(k) then 525
515 for i=1 to 8:if a2$=m$(i) then 535
520 next i
525 print"Hier ligt geen "a2$:return
530 print:print"Wat wil je pakken ?":return
535 print:print at$(K)" gepakt."
540 me$(i)=at$(k):at=at+1:at(k)=0:return
545 :
550 rem weg
555 if a2$="" then 575
560 for i=1 to 8:if a2$=me$(i) then 580
565 next i:print
570 print"Je hebt geen "a2$" bij je.":return
575 print:print"Wat wil je weggooien ?":return
580 if at(k)=1 then print"Dat kan nu
    niet.":return
585 print:print me$(i)" weggegooid."
590 me$(i)="":at=at-1
595 at$(k)=a2$:at(k)=1:return
600 :
```

Details

Nu even aandacht voor de specifieke programmaop - bouw. Als u die goed doorziet, is het een peuleschil om het avontuur te veranderen, uit te breiden, aan te passen enzovoort.
Het meest elementaire gegeven is de relatie-tabel. Deze tabel geeft de relatie van de verschillende lokatie ten aanzien van de onderlinge bereikbaarheid binnen het avontuur. Elke lokatie kan grenzen aan vier andere lokaties: NOORD, ZUID, OOST en WEST. Een slimme programmeur kan er zes van maken: OMHOOG en OMLAAG. Daar waar een aangrenzende lokatie niet bereikbaar is, is dus geen doorgang. Hoewel het best mogelijk is, dat de lokatie er wel aan grenst. De volgorde van de relaties is van belang. In diezelfde volgorde staan de grenslokaties in de tabel rl. Bekijk de plattegrond van dit avontuur en lees de relaties van de eerste lokatie: (afb.1)
De mogelijke doorgangen gaan naar NOORD: 2, ZUID:5, OOST:3 en als vierde WEST:6. Daar waar een nul staat, is geen doorgang mogelijk.
Neem bijvoorbeeld lokatie 2. Die heeft alleen doorgangen naar ZUID en WEST. NOORD en OOST zijn niet toegankelijk. Als de relatie-tabel is gelezen in regel 375 en legaal is bevonden, volgt een uitgang met de waarde 1 in variabele V. Daarna volgt de sprong in de gekozen richting. Regel 850 doet de laatste controle en de informatie over de nieuwe lokatie wordt op het scherm gezet.
Bij een ongeldige richting (V=0) krijgt de speler een passende mededeling door middel van de regels 795-805 en de programma controle gaat terug naar het controledeel van het avontuur.
Zo af en toe ziet u AF=1 of AF=5 staan, gevolgd door GOSUB 480. De waarde in AF wordt van de start-punten (SC=100) afgetrokken en er wordt steeds gekeken of er nog punten over zijn. Zo ja, wordt er doorgespeeld. Zijn de punten verbruikt, dan eindigt het spel door middel van een sprong naar regelnummer 735. Daar loopt het avontuur helaas slecht af.

De ontknoping

De attributen, die verspreid zijn over het avontuur, hebben een vooraf bepaalde plaats. De plaats in de array AT$ is tevens de start-plaats

```
605 rem herstarten
610 print:print"Einde avontuur...."
615 print"Je score is: "sc
620 print:print"Druk op een toets....."
625 get t$:if t$="" then 625
630 run
635 :
640 rem wachten
645 print:for t=1 to 1500:next t:return
650 :
655 rem attributen in de lokatie
660 if at(k)=0 then 670
665 print:print"Ik zie een ";at$(k)
670 print:return
675 :
680 rem succesvol einde
685 print:print"Dit is de plaats van de
    oude kaart."
690 print"Hier moet de schat verborgen
    zijn...."
695 gosub 640
700 print"Zou het werkelijk zo zijn ?"
705 print"Je schep stuit op iets hards."
710 gosub 640
715 print"Ja!!!! Het is de schat!!!"
720 print"Gefeliciteerd."
725 goto 605
730 :
735 rem slecht einde
740 print:print"Je was hier al eerder..."
745 gosub 640
750 print"Niet erg slim...."
755 print"Wat is dat voor geluid?"
760 gosub 640
765 print"Met een doffe klap komt een knuppel"
770 print"neer op je hoofd. Alles wordt zwart."
775 gosub 640
780 for t=55296 to 56295:poke t,0:next t
785 sc=0:goto 605
790 :
795 rem geen doorgang
800 print:print"Daar kun je niet langs...."
805 af=1:gosub 480:return
810 :
815 rem lokatie sprong
820 k=rl(k,i):if k=4 then 865
825 for lk=1 to 3:print lk$(k,lk):next lk
830 if k=4 then gosub 910
835 if e=1 then 680
840 if k=5 then pb=pb+1:if pb>int(rnd(1)*2)+2
    then 735
845 print:gosub 655
850 v=0:gosub 240:if v=0 then 850
855 goto 820
860 :
865 if me$(6)="boot" then 885
870 print"Hoe wou je daar naar toe ?"
875 print"Zwemmen zeker........"
880 k=3:goto 850
```

vervolg listing op pag. 58

in het avontuur. Door middel van de array AT wordt met een 0 of een 1 aangegeven of er in een lokatie een voorwerp ligt. Dit is van belang voor de subroutine op regelnummer 655. Zo kunnen voorwerpen verplaatst worden binnen de lokaties van het avontuur. Een voorwerp kan worden opgepakt in lokatie 6 en weer worden weggedaan in lokatie 1. Daar blijft het voorwerp dan liggen tot het weer wordt gepakt. Door de volgorde van de attributen en de bijbehorende getallen te veranderen, kunt u het avontuur anders inrichten. U kunt er ook attributen toevoegen of weghalen.

De extra komma's in regel 1400 en 1405 zijn geen vergissing. Ze moeten er echt staan, omdat in lokatie 2,4 en 7 geen voorwerp ligt. De elementen in de array AT$ blijven zo leeg, maar wel beschikbaar. Om dezelfde redenen staan er een aantal getallen in de DATA regels voor de informatie over de voorwerpen. Voorwerp 2, 4 en 7 bestaan niet en hebben dus geen informatie. Als u voorwerpen toevoegt, kunt u daar 4 regels informatie per voorwerp plaatsen. Als de speler zich op lokatie 4 bevindt, daar waar de schat 'verstopt' is, wordt de inventaris nagegaan. Voldoet deze aan een bepaalde norm, dan eindigt het programma succesvol door een sprong naar regel 680. De speler krijgt een pluim en de score. Als u wilt, kunt u de schat ook ergens anders verstoppen of zelf willekeurig ergens plaatsen. De vraag in regel 830: IF K=4 THEN GOSUB kunt u dan laten slaan op de gekozen lokatie. Zowel bij pech als bij succes kan opnieuw worden gespeeld door middel van de regels 605-630.

Tenslotte

Het avontuur, zoals u dat hier besproken zag, is gering van omvang. Maar wat weerhoudt u er van om het avontuur uit te breiden tot pak weg 20 lokaties?!

Toch bleek het voor de kinderen, die ik er mee liet spelen een boeiende uitdaging. In groepjes van steeds twee kinderen in de leeftijd van 11-12 jaar mochten een aantal kinderen op avontuur gaan. Ze waren zeer enthousiast over deze andere benadering van computergebruik in de klas. Ook de voorafgaande sceptische houding van enkele van hen veranderde zeer snel in een opgewonden spelen van het spel.

Ik wens alle onderwijsgevenden, die het avontuur aandurven veel plezier en vooral veel inventieve momenten bij het zelf maken van een avontuur.

●

B.M.

De redactie is erg benieuwd naar eventuele bedenkers van nieuwe adventures. Niet alleen voor de slimme spelers/sters, maar ook duidelijk adventure-leerprogramma's. Indien er voldoende reaktie komt zullen we een aantal artikelen speciaal met deze educatieve adventures uitbrengen. Alle ideeën zijn welkom!

KORTE PROGRAMMAOPBOUW

100	Copyright 1987
130	Initialisatie
220	Introductie
240	Antwoord-analyse
355	kijken zonder voorwerp
390	inventaris
425	help
445	kijk naar voorwerp
480	score bijhouden
500	pak voorwerp
550	weg voorwerp
605	herstarten
640	wachtlus
655	attributen op de lokatie
680	einde met succes
735	einde zonder succes
795	geen doorgang
815	lokatiesprong
910	attributen controle
1000	gegevens in dataregels

Print-out

Met o.a. Tour de France - Domino - Woorden - Mastermind C-16

Listingtelefoon

Voor vragen over de in Commodore Info verschenen listings blijft de listingtelefoon beschikbaar. Elke maandag zitten we van 17.00 tot 21.00 uur paraat voor vragen en opmerkingen op nummer 02155-25162. Denk er wel aan dat het niet mogelijk is om op andere tijden informatie over listings te krijgen, niet op bovenstaand nummer en ook niet bij de redactie.

Rob Goudriaan.

Inhoud van dit listingdeel

Syntax Checksum

Het overtikken van een listing kan een heel karwei zijn en als u een beetje normaal mens bent dan maakt u daarin beslist een aantal fouten. Nu is niets moeilijker om de fouten uit je eigen werk te halen. Al geruime tijd geleden heeft Jan Bodzinga hiervoor een zgn. **Checksum-programma** geschreven. Om de vele nieuwe lezers van Commodore-Info te helpen volgt hieronder nog een keer een volledige uitlegh over de werking van dit programma, waarmee het, hoe vreemd dat misschien ook lijkt, echt mogelijk is om met behulp van dit programma de fouten in elke door ons geplaatste listing op te sporen.

Hiervoor gaat u als volgt te werk:

1. U tikt de listing heel zorgvuldig over en SAVEt hem voordat u het programma RUNt op een diskette of een cassette.
2. U tikt het RUN commando in. Mocht het programma de boodschap **'Fout in dataregels!'** geven dan heeft u een fout bij het overtikken gemaakt. Herstel de fout en SAVE de verbeterde versie. Mocht het programma met de boodschap **'data is weggezet checksum testten met sys ....'** komen dan is tot dusver alles goed. Het programma is nu in een stukje machinetaalgeheugen gezet. Als u het NEW commando geeft blijft het toch in de computer staan.

Alle door ons geplaatste programma's zijn in Basic geschreven. Als u een programma heeft overgetikt SAVE het eerst, mocht er iets mis gaan dan hoeft niet de gehele listing opnieuw te gaan intikken. Als u nu een programma op fouten wilt gaan controleren dan kunt u dat in het geheugen laden (wel eerst het checksumprogramma hebben gerund). Vervolgens typt u zonder het programma te runnen de opdracht **sys 49152** (C-64) of **sys 1536** (C-16 en Plus/4) in.

Als alles goed is gegaan loopt er nu een rij regelnummers over het scherm met getallen erachter. Dezelfde lijst staat ook achter elk door ons geplaatste programma. Wijkt nu een nummer achter een regelnummer af van het nummer dat in het blad staat dan heeft u in die regel iets anders ingetikt dan er in het blad stond. U kunt de stroom getallen d.m.v. de RUN/STOP toets pauzeren en weer vervolgen met de F1 of F7 toets. Het is uitermate belangrijk dat u goed met dit programma overweg kunt en mocht U het niet goed werkend krijgen bel dan gerust even met onze listingservice telefoonlijn.

```
  1 rem ********************************
    **
  2 rem basic loader "SYNTAX.CHECKSUM"
  3 rem na de commando's 'run' en 'new'
  4 rem blijft dit programma in het ge-
  5 rem heugen. laad het te testen pro-
  6 rem gramma en tik daarna sys 49152.
  7 rem ********************************
    **
 10 i=49152  :rem beginadres
 20 reada:ifa<0then40:rem data ingeleze
    n
 30 pokei,a:i=i+1:b=b+a:goto20
 40 if b<>16844thenprint"[CLR-HOME]fout
     in dataregels!":b=0:end
 50 poke49184,148:poke49185,192
 55 i=49300
 60 read a: ifa<0then80
 70 pokei,a:b=a+b:i=i+1:goto60
 80 if b<>20068thenprint"[CLR-HOME]fout
     in dataregels! (vanaf regel 240)":
    b=0:end
 90 print"data is weggezet"
 95 print"checksum testen met sys49152"
100 data 165,43,166,44,133,163,134,164,
     169, 147
110 data 32,210,255,160,0,240,3,32,73,
     192
120 data 32,73,192,208,1,96,32,225,255,
      208
130 data 3,76,116,164,32,81,192,32,73,1
    92
140 data 240,12,201,32,240,247,24,101,1
    67,133
150 data 167,76,37,192,166,167,169,0,13
    2,168
160 data 32,205,189,169,13,32,210,255,1
    64, 168
170 data 76,17,192,200,208,2,230,164,17
    7,163
180 data 96,162,0,189,123,192,240,6,32,
    210
190 data 255,232,208,245,32,73,192,170,
    32,73
200 data 192,132,168,32,205,189,162,3,1
    69,32
210 data 32,210,255,202,208,250,169,0,1
    33,167
220 data 164,168,96,82,69,71,69,76,32,0
230 data -1
240 data 165,197,201,3,240,7,201,4,240
250 data 6,76,148,192,76,34,192,169
260 data 147,32,210,255,76,161,192
270 data -1

*** EINDE LISTING ***

    syntaxchecksum listtestprogramma

    regel 1   249
    regel 2   84
    regel 3   125
    regel 4   2
    regel 5   246
    regel 6   152
    regel 7   249
    regel 10   157
    regel 20   64
    regel 30   38
    regel 40   57
    regel 50   14
    regel 55   251
    regel 60   192
    regel 70   42
    regel 80   244
    regel 90   245
    regel 95   237
    regel 100   183
    regel 110   158
    regel 120   232
    regel 130   183
    regel 140   96
    regel 150   96
    regel 160   127
    regel 170   71
    regel 180   223
    regel 190   73
    regel 200   79
    regel 210   109
    regel 220   106
    regel 230   225
    regel 240   16
    regel 250   163
    regel 260   92
    regel 270   225

ready.
```

Tour de France

Met dit programma van J.Terpstra uit Maarssenbroek kun je gegevens vastleggen om bij een vereniging, club, werk of privé het Tour de France spel te spelen. De ene dag breng je de tips in, de volgende dag de uitslagen. Het programma is volledig menugestuurd en voorzien van een complete uitleg. De standen kunnen naar een cassette worden weggeschreven.

Wie maakt een programma om een klassement van de Tour de France bij te houden? Lieft met punten en de verschillende klassementen. En dit alles weer in en zo kort mogelijk programma. Ik wacht af.

```
10 rem ***       tour de france
        ***
15 rem * dit programma is gemaa
   kt door: *
20 rem *        johannes terpstr
   a            *
25 rem *        duivenkamp 483
                *
30 rem *     3607 bh maarssenbr
   oek          *
35 poke53280,0:poke53281,0:poke
   646,15:printchr$(14)
40 zz$="{HOME}{23xneer}":dimt$(
   23),r$(23,3),n$(18),k$(12)
45 print"{CLR-HOME}{4xneer}{2xspati
   e}wens je uitleg (j/n)":poke
   198,0:wait198,1:geta$
50 ifa$="j"then685
55 print"{CLR-HOME}{3xneer}{RVS-aan
   }{8xspatie}t o u r{2xspatie}
   d e{2xspatie}f r a n c e{7xspati
   e}"
60 print"{3xneer}{3xspatie}1){2xspa
   tie}laden van vorige gegeven
   s"
65 print"{neer}{3xspatie}2){2xspati
   e}invoeren tip nw. etappe"
70 print"{neer}{3xspatie}3){2xspati
   e}invoeren uitslag etappe"
75 print"{neer}{3xspatie}4){2xspati
   e}berekenen van de punten +"
80 print"{7xspatie}weergave sta
   nd"
85 print"{neer}{3xspatie}5){2xspati
   e}saven van de gegevens"
90 print"{neer}{3xspatie}6){2xspati
   e}stoppen"
95 print"{3xneer}{3xspatie}{RVS-aan
   }{2xspatie}wat is uw keuze?{3xsp
   atie}{RVS-uit}";:inputx
100 ifx<1orx>6thenprint"{5xop}":
    goto95
105 onxgosub155,225,265,315,610,
    115
110 goto55
115 rem *** einde ***
120 print"{CLR-HOME}{8xneer} ben
    t u niet vergeten de nieuwe"
125 print"{2xneer} gegevens vast
     te leggen?"
130 print"{2xneer} heeft u alle
    gegevens die nodig"
135 print"{2xneer} zijn om de ei
    nduitslag te bepalen"
140 print"{2xneer} ingevoerd????
    ?? (j/n)";:inputz$
145 ifz$="j"thenend
150 ifz$<>"j"then55
155 rem *** laden ***
160 poke53280,7
165 print"{CLR-HOME}{2xneer} {RVS-aa
    n}{5xspatie}laden van de vor
    ige gegevens{5xspatie}"
170 print"{2xneer}{2xspatie}is d
    e cassette in orde? (j/n)";:
    inputz$:ifz$="j"then185
175 print"{2xneer}{2xspatie}maak
     deze dan nu eerst in orde!!
    !!"
180 forx=1to4000:next:goto155
185 open1,1,0,"tour"
190 input#1,et
195 forx=1toet:input#1,t$(x):next
200 input#1,te
205 forx=1tote:fory=1to3:input#1
    ,r$(x,y):next:next
210 close1
215 print"{4xneer}{3xspatie}{RVS-aan
    }{2xspatie}alles is ingeleze
    n{3xspatie}"
220 return
225 rem *** invoer tip ***
230 poke53280,3
235 print"{CLR-HOME}{2xneer} {RVS-aa
    n}{18xspatie}tip{17xspatie}"
240 print"{3xneer}{2xspatie}welk
    e etappe?";:inputet
245 print"{neer}{2xspatie}welke
    renner tip je?";:inputt$(et)
250 printzz$"{2xspatie}alles goe
    d ingevuld? (j/n)";
255 inputz$:ifz$="n"then225
260 return
265 rem *** invoer uitslag ***
270 poke53280,5
275 print"{CLR-HOME}{2xneer} {RVS-aa
    n}{15xspatie}uitslag{16xspatie}"
280 print"{3xneer}{2xspatie}welk
    e etappe?";:inputte
285 print"{neer}{2xspatie}renner
     1 ste plaats:";:inputr$(te,
    1)
290 print"{neer}{2xspatie}renner
     2 de plaats:";:inputr$(te,2
    )
295 print"{neer}{2xspatie}renner
     3 de plaats:";:inputr$(te,3
    )
300 printzz$"{2xspatie}alles goe
    d ingevuld? (j/n)";
305 inputz$:ifz$="n"then265
310 return
```

```
315 rem *** berekenen v.d. punte
    n ***
320 rem * de regels 315 t/m 380
    kun je*
325 rem * gebruiken voor renners
     die  *
330 rem * een speciale behandeli
    ng    *
335 rem * krijgen bij het toeken
    nen   *
340 rem * van de punten omdat b.
    v een *
345 rem * ned. of een klimmer of
     een  *
350 rem * sprinter zijn. deze ge
    gevens*
355 rem * leg je vast in stringa
    ray's.*
360 rem * n$(1)t/m ... voor ned.
          *
365 rem * k$(1)t/m ... voor klim
    mers  *
370 rem * s$(1)t/m ... voor spri
    nters *
375 rem * et = teller bij de 'ti
    p'    *
380 rem * te = teller bij de uit
    slag  *
385 poke53280,8:print"{CLR-HOME}
     {RVS-aan}{15xspatie}punten{17xs
    patie}"
390 print"{2xspatie}etappe{4xspatie}
    punten{4xspatie}etappe{4xspatie}
    punten"
395 pu=0:a=0
400 forx=1tote
405 p=0
410 ift$(x)=r$(x,1)thenp=10:ifx=
    12orx=13orx=17orx=18thengosub
    580:goto440
415 ifp=10andx<>12andx<>13andx<>
    17andx<>18thengosub545:goto4
    40
420 ift$(x)=r$(x,2)thenp=8:ifx=1
    2orx=13orx=17orx=18thengosub
    590:goto440
425 ifp=8andx<>12andx<>13andx<>1
    7andx<>18thengosub555:goto44
    0
430 ift$(x)=r$(x,3)thenp=6:ifx=1
    2orx=13orx=17orx=18thengosub
    600:goto440
435 ifp=10andx<>12andx<>13andx<>
    17andx<>18thengosub565:goto4
    40
440 ift$(x)=""andr$(x,1)=""andr$
    (x,2)=""andr$(x,3)=""thenp=0
445 pu=pu+p
450 ifet=0then print"{2xneer} er
     is niets ingevoerd.":goto49
    5
455 ifx>9thena=a+1:ifa=1then475
460 ifx<10then470
465 goto480
470 printtab(3)x"{6xspatie}"p:goto
    485
475 print"{HOME}{neer}"
480 printtab(23)x"{6xspatie}"p
485 next
490 printzz$"{3xop}{3xspatie}{RVS-aa
    n}{2xspatie}totaal aantal pu
    nten ={2xspatie}{RVS-uit}";p
    u
495 printzz$;tab(5)" {RVS-aan}{2xspa
    tie}druk op een toets{2xspatie}"
    ;:geta$:ifa$=""then495
500 return
505 rem *** voor nederlandse ren
    ners***
510 fory=1to18:ift$(x)=n$(y)then
    p=p+2:return
515 next:return
520 fory=1to18:ift$(x)=n$(y)then
    p=p+2:return
525 next:return
530 fory=1to18:ift$(x)=n$(y)then
    p=p+2:return
535 next:return
540 rem *** voor sprinters ***
545 forz=1to8:ift$(x)=s$(z)thenp
    =p+2
550 next:gosub510:return
555 forz=1to8:ift$(x)=s$(z)thenp
    =p+2
560 next:gosub520:return
565 forz=1to8:ift$(x)=s$(z)thenp
    =p+2
570 next:gosub530:return
575 rem *** voor klimmers ***
580 forc=1to7:ift$(x)=k$(c)thenp
    =p+2
585 next:gosub510:return
590 forc=1to7:ift$(x)=k$(c)thenp
    =p+2
595 next:gosub520:return
600 forc=1to7:ift$(x)=k$(c)thenp
    =p+2
605 next:gosub530:return
610 rem *** saven v.d. gegevens
    ***
615 poke53280,11
620 print"{CLR-HOME}{2xneer} {RVS-aa
    n}{15xspatie}saven{18xspatie}"
625 print"{2xneer}{2xspatie}is d
    e cassette in orde? (j/n)";
630 inputz$:ifz$="j"then645
635 print"{2xneer}{3xspatie}maak
     deze dan nu eerst in orde!!
    !"
640 forx=1to4000:next:goto610
645 open1,1,1,"tour"
650 print#1,et
655 forx=1toet:print#1,t$(x):next
660 print#1,te
665 forx=1tote:fory=1to3:print#1
    ,r$(x,y):next:next
670 close1
```

```
675 print"{4xneer}{4xspatie}{RVS-aan
    }{2xspatie}alles is vastgele
    gd{2xspatie}"
680 return
685 rem *** uitleg ***
690 print"{CLR-HOME}{neer} {RVS-aan}
    {10xspatie}programma uitleg{12xs
    patie}"
695 print"{neer} met dit program
    ma kun je de"
700 print"{neer} gegevens vastle
    ggen waarmee je"
705 print"{neer} bij een krant o
    f club een tour"
710 print"{neer} de france wedst
    rijd speelt. "
715 print"{neer} de ene dag bren
    g je de tip in"
720 print"{neer} en de andere da
    g de uitslag. "
725 print"{neer} doe dit als vol
    gt: (tip)"
730 print"{neer} eerst het etapp
    e nr (et is de teller)"
735 print"{neer} dan de getipte
    renner (t$(et))"
740 printzz$;tab(3)"{2xspatie}{RVS-a
    an}{2xspatie}druk op een toe
    ts"
```

```
745 poke198,0:wait198,1
750 print"{CLR-HOME}{neer} als a
    lles goed is ingevuld"
755 print"{neer} beantwoord je d
    eze vraag met 'j'"
760 print"{neer} is het niet goe
    d ingevuld dan "
765 print"{neer} beantwoord je m
    et'n' en kun je"
770 print"{neer} je fout herstel
    len. "
775 print"{neer} bij 'j' krijg j
    e het menu weer"
780 print"{neer} terug op het sc
    herm en kun je b.v. "
785 print"{neer} de uitslag van
    de afgelopen "
790 print"{neer} etappe invullen
    "
795 printzz$;tab(3)"{2xspatie}{RVS-a
    an}{2xspatie}druk op een toe
    ts"
800 poke198,0:wait198,1
805 print"{CLR-HOME}{neer} de ui
    tslag breng je als volgt in:"
810 print"{neer} eerst het etapp
    e nr (te is de teller)"
815 print"{neer} dan renner 1 st
    e plaats (r$(te,1))"
```

```
820 print"{neer} dan renner 2 de
     plaats (r$(te,2))"
825 print"{neer} dan renner 3 de
     plaats (r$(te,3))"
830 print"{neer} voor het herste
    llen van fouten geldt"
835 print"{neer} hetzelfde als b
    ij de'tip'."
840 print"{neer} als het menu te
    rug is op het scherm"
845 print"{neer} kun je het volg
    ende onderdeel kiezen."
850 printzz$;tab(3)"{2xspatie}{RVS-a
    an}{2xspatie}druk op een toe
    ts"
855 poke198,0:wait198,1
860 print"{CLR-HOME}{neer} bij d
    e punten berekening is met e
    en"
865 print"{neer} aantal mogelijk
    heden rekening"
870 print"{neer} gehouden, deze
    zijn echter afhan-"
875 print"{neer} kelijk van het
    wedstrijd reglement."
880 print"{neer} de volgende mog
    elijkheden zijn er:"
885 print"{neer} a) alleen de ui
    tslag (r 410-435)"
890 print"{neer} b) speciale beh
    and. v. ned."
895 print"{neer}{4xspatie}(r 505
    -535)"
900 print"{neer} c) speciale beh
    and. sprinters"
905 print"{neer}{4xspatie}(r 540
    -570)"
910 print"{neer} d) speciale beh
    and. klimmers"
915 print"{neer}{4xspatie}(r 575
    -605)"
920 printzz$;tab(3)"{2xspatie}{RVS-a
    an}{2xspatie}druk op een toe
    ts"
925 poke198,0:wait198,1
930 print"{CLR-HOME}{neer} in de
     regel 315 t/m 380 is de"
935 print"{neer} ruimte om deze
    renner in stringaray's"
940 print"{neer} onder te brenge
    n."
945 print"{neer} n$(1) t/m .. is
     voor de ned."
950 print"{neer} s$(1) t/m .. is
     voor de sprinters"
955 print"{neer} k$(1) t/m .. is
     voor de klimmers"
960 print"{neer} als je deze voo
    rkeurs behandeling"
965 print"{neer} niet nodig hebt
     moet je deze"
970 print"{neer} routines verwij
    deren en de"
975 print"{neer} regel 410-435 a
    anpassen."
980 printzz$;tab(3)"{2xspatie}{RVS-a
    an}{2xspatie}druk op een toe
    ts"
985 poke198,0:wait198,1
990 print"{CLR-HOME}{neer} bij d
    e puntentelling is 'p' de"
995 print"{neer} teller voor de
    punten per etappe"
1000 print"{neer} en pu is de tot
    aal teller."
1005 print"{neer} deze p zul je m
    oeten aanpassen"
1010 print"{neer} aan het wedstri
    jd reglement."
1015 print"{neer} in de regels 41
    0, 420 en 430 is de "
1020 print"{neer} tweede 'if then
    ' bedoeld om te kijken"
1025 print"{neer} of de ingevoerd
    e etappe een berg"
1030 print"{neer} is of niet."
1035 print"{3xneer} veel succes m
    et dit programma"
1040 printzz$;tab(3)"{2xspatie}{RVS-a
    an}{2xspatie}druk op een toe
    ts"
1045 poke198,0:wait198,1
1050 goto55
```

*** EINDE LISTING ***

tour de france

REGEL 10	13	REGEL 165	150
REGEL 15	110	REGEL 170	216
REGEL 20	174	REGEL 175	127
REGEL 25	118	REGEL 180	62
REGEL 30	10	REGEL 185	67
REGEL 35	120	REGEL 190	122
REGEL 40	180	REGEL 195	241
REGEL 45	60	REGEL 200	122
REGEL 50	122	REGEL 205	254
REGEL 55	55	REGEL 210	209
REGEL 60	211	REGEL 215	225
REGEL 65	46	REGEL 220	142
REGEL 70	136	REGEL 225	75
REGEL 75	77	REGEL 230	248
REGEL 80	173	REGEL 235	145
REGEL 85	139	REGEL 240	25
REGEL 90	118	REGEL 245	71
REGEL 95	253	REGEL 250	53
REGEL 100	64	REGEL 255	202
REGEL 105	234	REGEL 260	142
REGEL 110	243	REGEL 265	119
REGEL 115	240	REGEL 270	250
REGEL 120	92	REGEL 275	189
REGEL 125	27	REGEL 280	25
REGEL 130	117	REGEL 285	139
REGEL 135	40	REGEL 290	42
REGEL 140	172	REGEL 295	44
REGEL 145	112	REGEL 300	53
REGEL 150	12	REGEL 305	206
REGEL 155	239	REGEL 310	142
REGEL 160	252	REGEL 315	234

```
REGEL 320    175      REGEL 640    58
REGEL 325    180      REGEL 645    68
REGEL 330    18       REGEL 650    142
REGEL 335    74       REGEL 655    5
REGEL 340    62       REGEL 660    142
REGEL 345    211      REGEL 665    18
REGEL 350    63       REGEL 670    209
REGEL 355    203      REGEL 675    38
REGEL 360    124      REGEL 680    142
REGEL 365    216      REGEL 685    85
REGEL 370    70       REGEL 690    3
REGEL 375    26       REGEL 695    97
REGEL 380    248      REGEL 700    195
REGEL 385    147      REGEL 705    248
REGEL 390    15       REGEL 710    209
REGEL 395    228      REGEL 715    37
REGEL 400    249      REGEL 720    85
REGEL 405    50       REGEL 725    139
REGEL 410    165      REGEL 730    136
REGEL 415    95       REGEL 735    217
REGEL 420    126      REGEL 740    61
REGEL 425    55       REGEL 745    96
REGEL 430    117      REGEL 750    235
REGEL 435    97       REGEL 755    137
REGEL 440    225      REGEL 760    235
REGEL 445    246      REGEL 765    4
REGEL 450    152      REGEL 770    223
REGEL 455    179      REGEL 775    170
REGEL 460    57       REGEL 780    187
REGEL 465    37       REGEL 785    143
REGEL 470    232      REGEL 790    72
REGEL 475    1        REGEL 795    61
REGEL 480    182      REGEL 800    96
REGEL 485    130      REGEL 805    93
REGEL 490    121      REGEL 810    136
REGEL 495    35       REGEL 815    123
REGEL 500    142      REGEL 820    26
REGEL 505    19       REGEL 825    28
REGEL 510    27       REGEL 830    49
REGEL 515    74       REGEL 835    48
REGEL 520    27       REGEL 840    73
REGEL 525    74       REGEL 845    38
REGEL 530    27       REGEL 850    61
REGEL 535    74       REGEL 855    96
REGEL 540    155      REGEL 860    237
REGEL 545    41       REGEL 865    168
REGEL 550    167      REGEL 870    18
REGEL 555    41       REGEL 875    239
REGEL 560    168      REGEL 880    141
REGEL 565    41       REGEL 885    172
REGEL 570    169      REGEL 890    248
REGEL 575    53       REGEL 895    245
REGEL 580    242      REGEL 900    58
REGEL 585    167      REGEL 905    243
REGEL 590    242      REGEL 910    213
REGEL 595    168      REGEL 915    250
REGEL 600    242      REGEL 920    61
REGEL 605    169      REGEL 925    96
REGEL 610    82       REGEL 930    57
REGEL 615    39       REGEL 935    103
REGEL 620    33       REGEL 940    46
REGEL 625    211      REGEL 945    126
REGEL 630    204      REGEL 950    72
REGEL 635    94       REGEL 955    218

REGEL 960    102      REGEL 1010   32
REGEL 965    158      REGEL 1015   118
REGEL 970    200      REGEL 1020   152
REGEL 975    131      REGEL 1025   171
REGEL 980    61       REGEL 1030   125
REGEL 985    96       REGEL 1035   111
REGEL 990    139      REGEL 1040   61
REGEL 995    5        REGEL 1045   96
REGEL 1000   6        REGEL 1050   243
REGEL 1005   82
                      READY.
```


nabestellen OUDE NUMMERS

Hieronder volgt een overzicht van de verschenen en nog beschikbare nummers.

Jaar	Aantal verschenen nummers	Nog beschikbaar
1984	5	niets
1985	10	alle nummers
1986	9	alle nummers

Reeds verschenen nummers zijn na te bestellen. Maar, helaas, NIET per brief, kaart of telefonisch. We leveren die oude nummers alleen bij vooruitbetaling op onze giro 1585491 t.n.v. Sala Communications/SAC te Blaricum of op onze bank in België BBL nr. 310050602562.

De prijs is *f* 6,75 per nummer (dus aangeven welk nummer).

Domino

Dit spel van Fons Reijsbergen is een computerversie van het spel dat we eigenlijk vroeger allemaal wel eens gespeeld hebben. De bedoeling is om als eerste al je stenen kwijt te raken door stenen met gelijke ogen aan elkaar te leggen. Het spel is voor een of twee spelers, als je alleen speelt neemt de computer drie spelers voor zijn rekening anders twee.

```
  1 rem domino / cbm-64
  2 rem door fons reijsbergen
  3 rem leidschendam / 070-27861
    9
  4 rem
  5 rem pas zelf de kleuren aan.
    ik werk op       een zwart/wi
    t tv.
  6 rem
 10 poke657,128:poke52,48:poke56
    ,48:clr
 20 rem*******hoofd programma
 30 gosub610:gosub1050:gosub800
 40 a$="66":r=1:k=1:print"{grijs 3}"
    :gosub1130:print"{WIT}"
 50 rem*******hoofd loop
 60 gosub890:ifa$(s,0)="c"thengosub
    1370:goto80
 70 gosub130
 80 ifk=0thenprintt$"{2xneer}{2xrech
    ts}PAS{7xspatie}":gosub1590:
    gosub460:printt$"{2xneer}{2xrech
    ts}{3xspatie}":ps=ps+1
 90 gosub490:ifw=1orps=4then520
100 s=s+1:ifs=5thens=1:ps=0
110 goto60
120 re*******speler beurt
130 gosub1560
140 i=1:i$="STEEN NR.{2xspatie}{2xli
    nks}":gosub250:k=val(i$):a$=
    a$(s,k):se=k:ifk=0thenreturn

150 i=2:i$="KANT a-b ":gosub250:
    k=1:ifi$="b"thenk=2
160 i=3:i$="RICHTING ":gosub250:
    r=val(i$)
170 p1=p(1,3):p2=p(2,3)
180 ifval(left$(a$,1))<>p(k,3)and
    val(right$(a$,1))<>p(k,3)then
    350
190 ifval(left$(a$,1))=p(k,3)then
    p(k,3)=val(right$(a$,1)):goto
    210
200 p(k,3)=val(left$(a$,1))
210 gosub1130:iff=1thenp(1,3)=p1
    :p(2,3)=p2:goto350
220 a$(s,se)="77"
230 return
240 rem--------input
250 printt$"{2xneer}{2xrechts}"i
    $;:poke204,0:poke198,0
260 geti$:ifi$=""then260
270 ifi=1andi$="p"theni$="0":goto
    320
280 ifi=1andi$>"0"andi$<"8"then3
    20
290 ifi=2and(i$="a"ori$="b")then
    320
300 ifi=3and(i$>"0"andi$<"5")then
    320
310 goto260
320 poke204,1:print"{links}{2xspatie
    }"
330 gosub1590:return
340 rem--------fout melding
350 printt$"{2xneer}{2xrechts}IN
    VOER{shift-SPATIE}FOUT "
360 gosub1620:gosub460:goto130
370 re*******print komputers keu
    s
380 k$=" A":ifk=2thenk$=" B"
390 printt$"{2xneer}{2xrechts}ST
    EEN NR."se:gosub440
400 printt$"{2xneer}{2xrechts}KA
    NT a-b "k$:gosub440
410 printt$"{2xneer}{2xrechts}RI
    CHTING "r:gosub440
420 printt$"{2xneer}{2xrechts}{13xsp
    atie}":gosub460
430 return
440 gosub1590:gosub460:return
450 rem******pauze
460 fori=1to800:next:return
470 fori=1to8000:next:return
480 rem******is s winnaar ?
490 w=1:fori=1to7:ifa$(s,i)<>"77"
    thenw=0
500 next:return
510 rem******print punten en win
    naar
520 gosub800:gosub460:print"{HOME}
    {2xneer}"tab(12)" DE PUNTEN
    {2xneer}":gosub1560:w=99:ww=
    s
530 fori=1to4:print"{2xrechts}"n
    $(i)tab(15)" = ";
540 t=0:forj=1to7
550 ifa$(i,j)<>"77"thent=t+val(left$
    (a$(i,j),1))+val(right$(a$(i
    ,j),1))
560 next:ift<wthenw=t:ww=i
570 printttab(22)"punten over ":
    nexti
580 printt$"{neer}{4xrechts}DE W
    INNAAR IS{2xspatie}"n$(ww)
590 gosub460:gosub1560:gosub470:
    gosub1560:run
600 rem******vragen aan speler
610 poke53280,0:poke53281,0:poke
    53272,23
620 print"{CLR-HOME}{WIT}{neer}{5xsp
    atie}************************
    *******"
630 print"{5xspatie}***********
    DOMINO ***********"
640 print"{5xspatie}************
    *******************"
650 printtab(13)"{2xneer}WACHT E
    VENTJES{op}"
660 gosub960:gosub1560
```

```
670 print"{3xspatie}Aantal spele
    rs 1 of 2 ? (d=demo) ";
680 geta$:ifa$=""thent=t+1:ift<7
    50then680
690 ifa$="d"ora$=""thenas=0:print
    a$:goto770
700 ifa$<>"1"anda$<>"2"then680
710 printa$:as=1:ifa$="2"thenas=
    2
720 input"{2xneer}Naam speler 1"
    ;n$(1):a$(1,0)="m"
730 print"{neer}Naam speler 2{2xspat
    ie}";n$(2)
740 ifas=2theninput"{neer}Naam s
    peler 3";n$(3):a$(3,0)="m":goto
    760
750 print"{neer}Naam speler 3{2xspat
    ie}";n$(3)
760 print"{neer}Naam speler 4{2xspat
    ie}";n$(4)
770 print"{4xneer}{3xspatie}WACH
    T NU EVENTJES"
780 return
790 rem******print border
800 a$="]":fori=1to38:a$=a$+" ":
    next:a$=a$+"]":b$="":fori=1to
    38:b$=b$+"@":next
810 print"{CLR-HOME}0"b$".";:for
    i=1to16:printa$;:next
820 print"+"b$"3]{GEEL} PAS = p{6xsp
    atie}RICHTING = {RVS-aan}8{RVS-u
    it}1 {RVS-aan}9{RVS-uit}2 {RVS-a
    an}:{RVS-uit}3 {RVS-aan};{RVS-ui
    t}4{2xspatie}{WIT}]";
830 print"+"b$"3";:fori=1to3:print
    a$;:next:print"-"b$"=";
840 ifas=0thenprint"{4xlinks}DEM
    O{op}DEMO"
850 ifas=0thenprint"{HOME}DEMO"tab(
    36)"DEMO"
860 print"{HOME}"
870 return
880 rem******print gegevens beur
    t s
890 printt$"{2xrechts}Speler "n$
    (s)"{10xspatie}"
900 printt$tab(25)"{grijs 3}1 2
    3 4 5 6 7{WIT}"tab(65);:poke
    199,2
910 rem========regel 885 voor bl
    anko                     compu
    terstenen
920 rem ifas<>0anda$(s,0)="c"the
    nfori=1to7:print" {neer}{links}
     {op}{rechts}";:next:goto900

930 fori=1to7:printleft$(a$(s,i)
    ,1)"{neer}{links}"right$(a$(
    s,i),1)"{op}{rechts}";:next
940 poke199,0:return
950 rem******set up start variab
    le
960 dimp(2,3),a$(4,7),s$(2,28)
970 fori=1to28:reads$(2,i):next:
    fori=1to2:forj=1to3:readp(i,
    j):nextj,i
980 fori=1to4:readn$(i):next
990 fori=49152to49204:reada:poke
    i,a:next:sys49152
1000 fori=15744to15839:reada:poke
     i,a:next
1010 poke53272,30:fori=1to4:a$(i,
     0)="c":next
1020 t$="{HOME}":fori=1to20:t$=t$
     +"{neer}":next
1030 return
1040 rem*****schud en deel stenen

1050 fori=1to28:s$(1,i)=s$(2,i):next

1060 fori=1to7:forj=1to4
1070 x=int(rnd(1)*28)+1:ifs$(1,x)
     ="77"then1070
1080 a$(j,i)=s$(1,x):s$(1,x)="77"
     :ifa$(j,i)="66"thens=j:a$(j,
     i)="77"
1090 nextj,i
1100 print"{2xneer}{3xrechts}"n$(
     s)" begint het spel":gosub46
     0:gosub460:poke198,0:gosub15
     60
1110 return
1120 rem*****plaats steen a$
1130 poke199,2:f=0:pl=k:gosub1340

1140 ifas=0thengetdb$:ifdb$<>""then
     run
1150 c$=right$(str$(p(k,3)),1):b$
     =left$(a$,1):ifb$=c$thenb$=right$
     (a$,1)
1160 x=p(k,1)*40+p(k,2)+1024
1170 onrgosub1210,1240,1270,1300
1180 poke199,0:pl=1:gosub1340:print
     "a":pl=2:gosub1340:print"b"
1190 return
1200 rem-------links
1210 ifpeek(x-2)=32andpeek(x-1)=3
     2thenprint"{links}"c$b$:p(k,
     2)=p(k,2)-2:return
1220 goto1320
1230 rem-------hoog
1240 ifpeek(x-80)=32andpeek(x-40)
     =32thenprintb$"{op}{links}"c
     $:p(k,1)=p(k,1)-2:return
1250 goto1320
1260 rem-------rechts
1270 ifpeek(x+2)=32andpeek(x+1)=3
     2thenprintb$c$:p(k,2)=p(k,2)
     +2:return
1280 goto1320
1290 rem--------neer
1300 ifpeek(x+80)=32andpeek(x+40)
     =32thenprintb$"{neer}{links}"
     c$:p(k,1)=p(k,1)+2:return
1310 rem-------kan steen niet pla
     atsen
1320 f=1:return
1330 rem*****print at pl
1340 print"{HOME}";:fori=1top(pl,
     1):print"{neer}";:next:fori=
```

```
      1top(pl,2):print"{rechts}";:
      next
1350  return
1360  rem*****computerzet s
1370  tt=1:p1=p(1,3):p2=p(2,3)
1380  K=0:iftt>7thengoto1460
1390  fori=ttto7:ifa$(s,i)="77"then
      1450
1400  ifas=0thengetdb$:ifdb$<>""then
      run
1410  ifval(left$(a$(s,i),1))=p(1,
      3)thenK=1:se=i:i=7:r=1:goto1
      450
1420  ifval(left$(a$(s,i),1))=p(2,
      3)thenK=2:se=i:i=7:r=1:goto1
      450
1430  ifval(right$(a$(s,i),1))=p(1
      ,3)thenK=1:se=i:i=7:r=2:goto
      1450
1440  ifval(right$(a$(s,i),1))=p(2
      ,3)thenK=2:se=i:i=7:r=2:goto
      1450
1450  nexti
1460  ifK=0thenp(1,3)=p1:p(2,3)=p2
      :return
1470  p(K,3)=val(left$(a$(s,se),1)
      ):ifr=1thenp(K,3)=val(right$
      (a$(s,se),1))
1480  a$=a$(s,se):r=int(rnd(1)*4)+
      1:t=0
1490  gosub1130:iff=0thena$(s,se)=
      "77":gosub380:return
1500  r=r+1:ifr=5thenr=1:t=t+1:ift
      =2thentt=se+1:goto1380
1510  goto1490
1520  rem*****sid schoon
1530  si=54272:fortt=0to23:poKesi+
      tt,.:next:v=si+24:ad=si+5:sr
      =si+6:su=si+15
1540  sl=si+14:fu=si+1:w=si+4:poKe
      v,15:return
1550  rem*****bel
1560  gosub1530:poKead,10:poKesr,7
      5:poKesu,10:poKesl,10:poKefu
      ,4:poKew,21
1570  fortt=1to50:next:poKew,20:for
      tt=1to500:next:return
1580  rem*****piep
1590  gosub1530:poKesr,240:poKes,1
      5:poKefu,67:poKew,17:fortt=1
      to100:next
1600  poKew,16:return
1610  rem*****phaser
1620  gosub1530:poKead,136:poKesr,
      255:poKew,17:fortt=1to10:for
      zz=250to0step-20
1630  poKefu,zz:nextzz,tt:poKew,16
      :poKev,0:return
2000  rem*****data
2010  rem-------steen waarde
2020  data00,01,02,03,04,05,06,11,
      12,13,14,15,16,22,23,24,25,2
      6
2030  data33,34,35,36,44,45,46,55,
      56,66
2040  data10,19,6,10,20,6
2050  rem-------Komp.namen
2060  data"C.B.M","Commodore","Dom
      ino","Kompu"
2070  rem-------verpl.progr.mc
2080  data169,0,141,14,220,169,51,
      133,1,169,0,133,251,133,253,
      169,48,133
2090  data252,169,208,133,254,160,
      0,177,253,145,251,200,208,24
      9,230,254
2100  data230,252,169,80,197,252,2
      08,239,169,55,133,1,169,1,14
      1,14,220,96,234
2110  rem-------domino stenen grap
      h
2120  data255,255,255,255,255,255,
      255,255,255,255,255,239,255,
      255,255,255
2130  data255,255,255,219,255,255,
      255,255,255,191,255,239,255,
      251,255,255
2140  data255,219,255,255,255,219,
      255,255,255,219,255,239,255,
      219,255,255
2150  data255,219,255,219,255,219,
      255,255,0,0,0,0,0,0,0,0
2160  data0,16,48,255,48,16,0,0,0,
      16,56,124,16,16,16,16
2170  data0,8,12,255,12,8,0,0,16,1
      6,16,16,124,56,16,0

*** EINDE LISTING ***
```

REGEL 1	237	REGEL 270	171
REGEL 2	41	REGEL 280	127
REGEL 3	36	REGEL 290	237
REGEL 4	143	REGEL 300	207
REGEL 5	24	REGEL 310	33
REGEL 6	143	REGEL 320	62
REGEL 10	122	REGEL 330	36
REGEL 20	203	REGEL 340	53
REGEL 30	16	REGEL 350	98
REGEL 40	248	REGEL 360	14
REGEL 50	95	REGEL 370	230
REGEL 60	187	REGEL 380	232
REGEL 70	33	REGEL 390	117
REGEL 80	99	REGEL 400	93
REGEL 90	160	REGEL 410	186
REGEL 100	206	REGEL 420	18
REGEL 110	239	REGEL 430	142
REGEL 120	152	REGEL 440	133
REGEL 130	89	REGEL 450	16
REGEL 140	175	REGEL 460	109
REGEL 150	174	REGEL 470	157
REGEL 160	66	REGEL 480	201
REGEL 170	4	REGEL 490	53
REGEL 180	186	REGEL 500	74
REGEL 190	101	REGEL 510	149
REGEL 200	238	REGEL 520	255
REGEL 210	70	REGEL 530	208
REGEL 220	49	REGEL 540	249
REGEL 230	142	REGEL 550	110
REGEL 240	135	REGEL 560	140
REGEL 250	235	REGEL 570	124
REGEL 260	117		

REGEL	580	0
REGEL	590	115
REGEL	600	233
REGEL	610	138
REGEL	620	114
REGEL	630	63
REGEL	640	201
REGEL	650	27
REGEL	660	191
REGEL	670	149
REGEL	680	175
REGEL	690	235
REGEL	700	252
REGEL	710	32
REGEL	720	115
REGEL	730	184
REGEL	740	114
REGEL	750	186
REGEL	760	188
REGEL	770	31
REGEL	780	142
REGEL	790	214
REGEL	800	189
REGEL	810	212
REGEL	820	67
REGEL	830	115
REGEL	840	212
REGEL	850	91
REGEL	860	240
REGEL	870	142
REGEL	880	65
REGEL	890	52
REGEL	900	216
REGEL	910	196
REGEL	920	38
REGEL	930	31
REGEL	940	94
REGEL	950	240
REGEL	960	237
REGEL	970	191
REGEL	980	14
REGEL	990	74
REGEL	1000	116
REGEL	1010	114
REGEL	1020	5
REGEL	1030	142
REGEL	1040	82
REGEL	1050	64
REGEL	1060	72
REGEL	1070	83
REGEL	1080	100
REGEL	1090	65
REGEL	1100	162
REGEL	1110	142
REGEL	1120	10
REGEL	1130	92
REGEL	1140	59
REGEL	1150	252
REGEL	1160	200
REGEL	1170	13
REGEL	1180	216
REGEL	1190	142
REGEL	1200	75
REGEL	1210	10
REGEL	1220	79
REGEL	1230	247
REGEL	1240	2
REGEL	1250	79
REGEL	1260	147
REGEL	1270	38
REGEL	1280	79
REGEL	1290	244
REGEL	1300	127
REGEL	1310	171
REGEL	1320	241
REGEL	1330	31
REGEL	1340	192
REGEL	1350	142
REGEL	1360	22
REGEL	1370	201
REGEL	1380	125
REGEL	1390	23
REGEL	1400	59
REGEL	1410	130
REGEL	1420	132
REGEL	1430	132
REGEL	1440	134
REGEL	1450	203
REGEL	1460	43
REGEL	1470	78
REGEL	1480	144
REGEL	1490	65
REGEL	1500	122
REGEL	1510	87
REGEL	1520	11
REGEL	1530	156
REGEL	1540	56
REGEL	1550	52
REGEL	1560	205
REGEL	1570	138
REGEL	1580	143
REGEL	1590	60
REGEL	1600	73
REGEL	1610	36
REGEL	1620	249
REGEL	1630	92
REGEL	2000	123
REGEL	2010	253
REGEL	2020	125
REGEL	2030	41
REGEL	2040	89
REGEL	2050	158
REGEL	2060	60
REGEL	2070	201
REGEL	2080	111
REGEL	2090	73
REGEL	2100	125
REGEL	2110	207
REGEL	2120	217
REGEL	2130	212
REGEL	2140	217
REGEL	2150	119
REGEL	2160	30
REGEL	2170	174

READY.

Lichtkrant

Na het opstarten van dit spel moet U ongeveer 45 seconden wachten voor dat de lichtkrant over het scherm gaat lopen. De maker van dit spel Hans Simonz heeft er al een tekst ingezet. Door een druk op de vuurknop van de joystick krijgt U een menu op het scherm. U kunt nu de kleur veranderen of er een andere tekst inzetten.

```
10 poke52,128:poke56,128:clr:no
   =0:sc=0:te=1
11 dimdx(39,20),lx(39),a(255),t
   $(1000):print"wacht u even?"
12 forn=0to25:a(65+n)=n:next:for
   n=1to9:a(48+n)=n+25:next
13 a(48)=14:a(46)=35:a(33)=36:a
   (58)=37:a(39)=38:a(32)=39
14 mc=32973
15 forn=0to39:form=0to19
16 reada:ch=ch+a:dx(n,m)=a:next
   m,n:ifch<>78533thenprint"err
   or in data's":end
17 reada$:forn=1to40:lx(n-1)=val
   (mid$(a$,n,1)):next
18 forn=32768to33014:reada:poke
   n,a:ch=ch+a:next
19 ifch<>100060thenprint"error
   in data's":end
20 a$="'big lichtkrant'... gepr
   ogrammeerd in 1986 door hans
    siemons...  "
21 a$=a$+"druk 'vuur' voor star
   t!{3xspatie}"
22 forn=1to90:t$(n)=mid$(a$,n,1
   ):next:xx=90:gosub49:sysmc,0
   ,0,1
23 print"{CLR-HOME}wat wilt u?"
   :poke53270,200
24 print"1. kleur veranderen"
25 print"2. lichtkrant bekijken"
26 print"3. lichtkrant invoeren"
```

```
27 geta$:ifa$=""then27
28 ifa$="3"then39
29 ifa$="2"thensysmc,bo,sc,te:goto
   23
30 ifa$<>"1"then27
31 print"{CLR-HOME}1. randkleur
   {7xspatie}{4xlinks}"bo
32 print"2. schermkleur{5xspatie}
   {4xlinks}"sc
33 print"3. letterkleur{5xspatie}
   {4xlinks}"te:print"{neer}ret
   urn voor einde
34 geta$:ifa$=chr$(13)then23
35 ifa$="1"thenbo=(bo+1)and15:goto
   31
36 ifa$="2"thensc=(sc+1)and15:goto
   31
37 ifa$="3"thente=(te+1)and15:goto
   31
38 goto34
39 printchr$(5)chr$(147)"big li
   chtkrant":poke53280,0:poke53
   281,0
40 a$="":print"uw tekst: *"chr$
   (157);:xx=0:poke53270,200
41 getb$:ifb$=""then41
42 ifb$=chr$(20)andxx<>0then47
43 ifb$=chr$(13)andxx<>0then48
44 ifb$="a"then46
45 ifa(asc(b$))=0then41
46 xx=xx+1:t$(xx)=b$:printb$"*"
   chr$(157);:goto41
47 xx=xx-1:printchr$(157)"{2xspatie
   }"chr$(157)chr$(157)"*"chr$(
   157);:goto41
48 print" ":printchr$(17)"bedan
   kt, wacht u even":gosub49:goto
   23
49 ad=40960:forn=1toxx:x$=t$(n)
50 a=a(asc(x$))
51 forx=0tol x(a)-1:fory=0to4:poke
   ad,dx(a,y+x*5):ad=ad+1:nexty
   ,x
52 forx=0to4:pokead+x,32:next:a
   d=ad+5
53 next:pokead,0:return
54 data233,160,160,160,160,160,
   32,160,32,32,223,160,160,160
   ,160,32,32,32,32,32
55 data160,160,160,160,160,160,
   32,160,32,160,223,160,160,16
   0,105,32,32,32,32,32
56 data233,160,160,160,95,160,3
   2,32,32,160,223,95,32,233,10
   5,32,32,32,32,32
57 data160,160,160,160,160,160,
   32,32,32,160,223,160,160,160
   ,105,32,32,32,32,32
58 data160,160,160,160,160,160,
   32,160,32,160,160,32,32,32,1
   60,32,32,32,32,32
59 data160,160,160,160,160,160,
   32,160,32,32,160,32,32,32,32
   ,32,32,32,32,32
60 data233,160,160,160,95,160,3
   2,32,32,160,160,32,160,233,1
   05,223,32
61 data160,160,160
62 data160,160,160,160,160,32,3
   2,160,32,32,160,160,160,160,
   160,32,32,32,32,32
63 data160,160,160,160,160,32,3
   2,32,32,32,32,32,32,32,32,32
   ,32,32,32,32
64 data32,32,32,95,32,32,32,32,
   223,95,32,32,32,233,160,160,
   160,160,160,105
65 data160,160,160,160,160,32,2
   33,160,95,32,233,105,32,223,
   95,105,32,32,32,223
66 data160,160,160,160,160,32,3
   2,32,32,160,32,32,32,32,160,
   32,32,32,32,32
67 data160,160,160,160,160,223,
   95,32,32,32,233,105,32,32,32
   ,160,160,160,160,160
68 data160,160,160,160,160,223,
   95,32,32,32,32,223,95,32,32,
   160,160,160,160,160
69 data233,160,160,160,95,160,3
   2,32,32,160,223,160,160,160,
   105,32,32,32,32,32
70 data160,160,160,160,160,160,
   32,160,32,32,223,160,105,32,
   32,32,32,32,32,32
71 data233,160,160,160,95,160,3
   2,32,108,160,223,160,160,160
   ,127,32,32,32,32,32
72 data160,160,160,160,160,160,
   32,160,223,95,223,160,105,32
   ,223,32,32,32,32,32
73 data233,160,160,32,95,160,32
   ,160,32,160,223,32,223,160,1
   05,32,32,32,32,32
74 data160,32,32,32,32,160,160,
   160,160,160,160,32,32,32,32,
   32,32,32,32,32
75 data160,160,160,160,95,32,32
   ,32,233,105,160,160,160,160,
   160,32,32,32,32,32
76 data160,160,160,95,32,32,32,
   32,223,95,32,32,32,233,105,1
   60,160,160,105,32
77 data160,160,160,160,95,32,32
   ,32,233,105,32,32,32,223,95,
   160,160,160,160,105
78 data223,95,32,233,105,32,223
   ,160,105,32,32,233,160,95,32
   ,233,105,32,223,95
79 data95,32,32,32,32,223,160,2
   51,225,225,233,160,236,97,97
   ,105,32,32,32,32
80 data160,32,32,233,160,160,32
   ,233,105,160,160,233,105,32,
   160,160,105,32,32,160
81 data233,105,32,32,32,160,160
   ,160,160,160,32,32,32,32,32,
   32,32,32,32,32
82 data233,32,233,160,160,160,3
   2,160,32,160,223,160,105,32,
   160,32,32,32,32,32
83 data233,32,32,32,95,160,32,1
```

```
     60,32,160,223,160,160,160,10
     5,32,32,32,32,32
  84 data160,160,160,32,32,32,32,
     160,32,32,32,160,160,160,160
     ,32,32,160,32,32
  85 data160,160,95,32,95,160,32,
     160,32,160,160,32,223,160,10
     5,32,32,32,32,32
  86 data233,160,160,160,95,160,3
     2,160,32,160,223,95,223,160,
     105,32,32,32,32,32
  87 data160,32,32,32,32,160,103,
     118,225,245,160,160,231,246,
     97,32,32,32,32,32
  88 data233,160,160,160,95,160,3
     2,160,32,160,223,160,160,160
     ,105,32,32,32,32,32
  89 data233,160,95,223,95,160,32
     ,160,32,160,223,160,160,160,
     105,32,32,32,32,32
  90 data32,32,32,32,160,32,32,32
     ,32,32,32,32,32,32,32,32,32,
     32,32,32
  91 data160,160,160,32,160,32,32
     ,32,32,32,32,32,32,32,32,32,
     32,32,32,32
  92 data160,32,32,32,160,32,32,3
     2,32,32,32,32,32,32,32,32,32
     ,32,32,32
  93 data32,225,32,32,32,160,97,3
     2,32,32,32,32,32,32,32,32,32
     ,32,32,32
  94 data32,32,32,32,32,32,32,32,
     32,32,32,32,32,32,32,32,32,3
     2,32,32
  95 data3333334314434433333334444
     4423343333311121
  96 data32,32,32,32,3,32,32,32,3
     2,15,32,32,8,32,13
  97 data40,32,46,32,13,3,32,32,3
     2,15,41,32,19,32,4
  98 data32,32,9,32,15,49,32,5,32
     ,18,57,32,13,32,5
  99 data56,32,15,32,45,54,32,14,
     32,9,32,32,19,32,14
 100 data32,32,32,32,6,32,32,32,3
     2,15,32,32,32,32,32
 101 data32,32,32,32,32,0,160,0,1
     77,251,230,251,208,2,230
 102 data252,96,162,7,173,18,208,
     208,251,173,17,208,48,246,14
     2
 103 data22,208,202,16,240,162,7,
     142,22,208,162,0,189,1,4
 104 data157,0,4,189,41,4,157,40,
     4,189,81,4,157,80,4
 105 data189,121,4,157,120,4,189,
     161,4,157,160,4,232,224,40
 106 data208,221,32,81,128,168,20
     8,11,169,0,133,251,169,160,1
     33
 107 data252,76,152,128,141,39,4,
     32,81,128,141,79,4,32,81
 108 data128,141,119,4,32,81,128,
     141,159,4,32,81,128,141,199
 109 data4,173,0,220,41,16,208,14
     5,88,96,169,0,133,251,169
 110 data128,133,252,120,32,241,1
     83,142,32,208,32,241,183,142
     ,33
 111 data208,32,241,183,142,134,2
     ,32,68,229,169,54,133,1,32
 112 data92,128,169,55,133,1,96
 113 rem (c)mcmlxxxvi by h. siemo
     ns

*** EINDE LISTING ***
```

lichtkrant

REGEL 10	191	REGEL 63	165
REGEL 11	90	REGEL 64	77
REGEL 12	109	REGEL 65	21
REGEL 13	171	REGEL 66	9
REGEL 14	74	REGEL 67	62
REGEL 15	185	REGEL 68	21
REGEL 16	202	REGEL 69	218
REGEL 17	20	REGEL 70	158
REGEL 18	97	REGEL 71	18
REGEL 19	148	REGEL 72	11
REGEL 20	127	REGEL 73	168
REGEL 21	53	REGEL 74	9
REGEL 22	139	REGEL 75	218
REGEL 23	192	REGEL 76	176
REGEL 24	169	REGEL 77	20
REGEL 25	116	REGEL 78	236
REGEL 26	152	REGEL 79	149
REGEL 27	54	REGEL 80	103
REGEL 28	44	REGEL 81	9
REGEL 29	89	REGEL 82	210
REGEL 30	217	REGEL 83	118
REGEL 31	124	REGEL 84	109
REGEL 32	140	REGEL 85	126
REGEL 33	47	REGEL 86	227
REGEL 34	106	REGEL 87	180
REGEL 35	250	REGEL 88	12
REGEL 36	5	REGEL 89	227
REGEL 37	12	REGEL 90	221
REGEL 38	240	REGEL 91	115
REGEL 39	3	REGEL 92	15
REGEL 40	89	REGEL 93	28
REGEL 41	52	REGEL 94	171
REGEL 42	34	REGEL 95	122
REGEL 43	37	REGEL 96	119
REGEL 44	57	REGEL 97	124
REGEL 45	136	REGEL 98	93
REGEL 46	119	REGEL 99	190
REGEL 47	170	REGEL 100	168
REGEL 48	239	REGEL 101	160
REGEL 49	249	REGEL 102	207
REGEL 50	24	REGEL 103	175
REGEL 51	193	REGEL 104	212
REGEL 52	13	REGEL 105	34
REGEL 53	252	REGEL 106	235
REGEL 54	4	REGEL 107	203
REGEL 55	52	REGEL 108	95
REGEL 56	128	REGEL 109	252
REGEL 57	2	REGEL 110	12
REGEL 58	159	REGEL 111	36
REGEL 59	59	REGEL 112	210
REGEL 60	40	REGEL 113	34
REGEL 61	160		
REGEL 62	209		

READY.

Romeinse cijfers.

Weet U wat het romeinse getal MCMLXXXVII in ons tientallig stelsel is? Als U het programma dat ons door Erik Huynen uit Margraten is toegestuurd, heeft ingetikt heeft U daar nooit meer moeite mee. Hij heeft het programma op XXII-VIII-MCMLXXXVI naar ons toegestuurd en het is ongeveer MMCDXCIX karakters lang.

```
100 rem romeinse cijfers voor c-
    64
110 rem erik huynen
120 rem de haymestraat 6
130 rem 6269 cs margraten
140 rem tel. 04458-2287
150 print"{WIT}";:poke53281,6:poke
    53280,14:poke650,250:printchr$
    (142)
160 print"{CLR-HOME}{neer}"tab(4
    )"{RVS-aan} r o m e i n s e{2xsp
    atie}c i j f e r s "
170 printtab(7)"{neer}(r)omeins
    naar tientallig"
180 printtab(7)"(t)ientallig naa
    r romeins"
190 printtab(7)"(e)inde{3xneer}"

200 printtab(13)"{op}maak uw keu
    ze":forq=1to200:nextq
210 printtab(13)"{op}{RVS-aan}ma
    ak uw keuze":forq=1to200:next
    q
220 getc$:ifc$=""then200
230 ifc$="t"thenmo$="0123456789"
    :t$="tientallig-romeins":ma=
    38:gosub780:goto570
240 ifc$="e"thenclr:end
250 ifc$<>"r"then220
260 mo$="ivxlcdm":t$="romeins-ti
    entallig":ma=254:re=0:wa=0:gosub
    780
270 x=1
280 re=re+wa
290 ifx>len(ge$)then350
300 c1$=mid$(ge$,x,1):c2$=mid$(g
    e$,x+1,1)
310 fora=1to7
320 if c1$=mid$(mo$,a,1)then340
330 nexta
340 onagoto410,460,470,510,520,5
    50,560
350 printge$" =";re
360 print"{neer}heeft u nog een
    romeins getal? (j/n)"
370 geta$:ifa$=""then370
380 ifa$="j"thenre=0:wa=0:gosub7
    80:goto270
390 ifa$<>"n"then370
400 goto160
410 ifc2$="v"thenwa=4     :x=x+2 :
    goto280
420 ifc2$="x"thenwa=9     :x=x+2 :
    goto280
430 ifc2$="l"thenwa=49    :x=x+2 :
    goto280
440 ifc2$="c"thenwa=99    :x=x+2 :
    goto280
450 :                wa=1    :x=x+1 :
    goto280
460 :                wa=5    :x=x+1 :
    goto280
470 ifc2$="l"thenwa=40   :x=x+2 :
    goto280
480 ifc2$="c"thenwa=90   :x=x+2 :
    goto280
490 ifc2$="d"thenwa=490  :x=x+2 :
    goto280
500 :                wa=10   :x=x+1 :
    goto280
510 :                wa=50   :x=x+1 :
    goto280
520 ifc2$="d"thenwa=400  :x=x+2 :
    goto280
530 ifc2$="m"thenwa=900  :x=x+2 :
    goto280
540 :                wa=100  :x=x+1 :
    goto280
550 :                wa=500  :x=x+1 :
    goto280
560 :                wa=1000:x=x+1 :
    goto280
570 printge$" = ";
580 if len(ge$)<4then 610
590 l=val(left$(ge$,(len(ge$)-3)
    )):ifl=0then610
600 forq=1tol:print"m";:nextq
610 a1$(1)="c":a2$(1)="d":a3$(1)
    ="m":a1$(2)="x":a2$(2)="l":a
    3$(2)="c"
620 a1$(3)="i":a2$(3)="v":a3$(3)
    ="x"
630 r$=right$(ge$,3):ifr$="099"or
    r$="99"thenr$="100":printa1$
    (3);
640 forte=(4-len(r$))to3
650 ho(te)=val(mid$(r$,te+len(r$
    )-3,1))
660 onho(te)gosub 730,730,730,74
    0,750,760,760,760,770
670 nextte
680 print:print"{neer}heeft u no
    g een tientallig getal? (j/n
    )"
690 getq$:ifq$=""then690
700 ifq$="j"thengosub780:goto570

710 ifq$<>"n"then690
720 goto160
730 forq=1toho(te):printa1$(te);
    :nextq:return
740 printa1$(te);a2$(te);:return

750 printa2$(te);:return
760 printa2$(te);:forq=1to(ho(te
    )-5):printa1$(te);:nextq:return

770 printa1$(te);a3$(te);:return

780 ge$=""
790 print"{CLR-HOME}":printtab(5
    )"{RVS-aan}{5xspatie}"t$"{4xspat
    ie}"
800 printtab(5)"{RVS-aan} tik he
```

```
     t getal in en sluit "
810 printtab(5)"{RVS-aan} af met
      <return>:{10xspatie}{2xneer}"
820 print"{RVS-aan} {links}";
830 getin$:ifin$=""then830
840 ifin$=chr$(13)andnotge$=""then
     print"{RVS-uit} ":print"{neer}
     even geduld hoor...{neer}":return
850 fora=1tolen(mo$)
860 ifmid$(mo$,a,1)=in$then880
870 nexta:goto830
880 print"{RVS-uit}";in$;
890 ge$=ge$+in$:if len(ge$) <> m
     a then820
900 print:print"{neer}maximale l
     engte...":print"{neer}even g
     eduld hoor...{neer}":return

*** EINDE LISTING ***
```

romeinse cijfers

REGEL 100	23	REGEL 520	184
REGEL 110	145	REGEL 530	198
REGEL 120	145	REGEL 540	233
REGEL 130	157	REGEL 550	237
REGEL 140	167	REGEL 560	25
REGEL 150	104	REGEL 570	5
REGEL 160	63	REGEL 580	116
REGEL 170	110	REGEL 590	45
REGEL 180	93	REGEL 600	81
REGEL 190	201	REGEL 610	58
REGEL 200	26	REGEL 620	158
REGEL 210	44	REGEL 630	126
REGEL 220	99	REGEL 640	93
REGEL 230	47	REGEL 650	14
REGEL 240	42	REGEL 660	121
REGEL 250	39	REGEL 670	27
REGEL 260	142	REGEL 680	239
REGEL 270	59	REGEL 690	140
REGEL 280	34	REGEL 700	114
REGEL 290	151	REGEL 710	60
REGEL 300	2	REGEL 720	32
REGEL 310	128	REGEL 730	61
REGEL 320	184	REGEL 740	216
REGEL 330	195	REGEL 750	29
REGEL 340	143	REGEL 760	253
REGEL 350	156	REGEL 770	217
REGEL 360	76	REGEL 780	166
REGEL 370	103	REGEL 790	86
REGEL 380	198	REGEL 800	225
REGEL 390	39	REGEL 810	19
REGEL 400	32	REGEL 820	199
REGEL 410	106	REGEL 830	20
REGEL 420	113	REGEL 840	45
REGEL 430	153	REGEL 850	29
REGEL 440	149	REGEL 860	228
REGEL 450	137	REGEL 870	33
REGEL 460	141	REGEL 880	160
REGEL 470	144	REGEL 890	51
REGEL 480	140	REGEL 900	254
REGEL 490	193		
REGEL 500	185		
REGEL 510	189	READY.	

Woorden.

Dit programma is gemaakt door Klaster-soft uit Roden. Het programma is gemaakt om woorden te leren schrijven. Je krijgt een woord te zien wat je goed moet bekijken. Als het van het scherm verdwenen is moet je het natypen. De tijd die het woord op het scherm blijft staan is zelf in te stellen.

Hoe korter het woord blijft staan hoe moeilijker het spel wordt.

```
  10 rem ************************
     *******
  20 rem *
          *
  30 rem * software voor de commo
     dore  *
  40 rem *          64
          *
  50 rem *   dit is software voor
      de   *
  60 rem *       prijs vraag van d
     e  '  *  70 rem *         com
     modore info no5    *
  75 rem *-----------------------
     ------*
  76 rem *
          *
  77 rem * dit programma is gemaa
     kt    *
  78 rem * door klaster-soft
          *
  79 rem * havixhorst 37
          *
  80 rem * 9301 wz roden
          *
  81 rem *-----------------------
     ------*
  82 rem *
          *
  83 rem * geschikt voor een kleu
     ren   *
  84 rem *         beeldbuis
          *
  85 rem *
          *
  86 rem ************************
     *******
 100 gosub 5000:rem ***5000-5499
     titelscherm            ***
 500 gosub 5500:rem ***5500-5999
     vullentabel            ***
1000 gosub 6000:rem ***6000-6499
     introduktie          ***
1200 if f3=1then run 100
1500 gosub 6500:rem ***6500-6999
     bepaal duur en woord***
2000 gosub 7000:rem ***7000-7499
     laat het woord zien ***
2500 gosub 7500:rem ***7500-7999
     input routine          ***
3000 gosub 8000:rem ***8000-8499
     beoordeel antwoord  ***
3010 iff1=1then f1=0:goto 2000
3020 iff2=1then f2=0:goto 1500
```

```
3500 gosub 8500:rem ***8500-8999
     eindroutine          ***
4000 end
5000 rem ** titelscherm ***
5010 poke 53280,0:poke53281,0:print
     chr$(147):f3=0
5012 rem ************************
     *****
5013 rem * 5020 t/m 5090 print wo
     orden
5014 rem ************************
     *****
5020 dim ar(98)
5030 for x=1 to 98
5040 read ar(x)
5050 next
5060 printchr$(147)
5070 y=y+1:ify>98 then 5120
5080 poke ar(y),81:poke ar(y)+542
     72,6
5090 goto 5070
5095 rem ************************
     ****
5096 rem * 5110 t/m 5300 print ra
     nd *
5097 rem ************************
     ****
5120 for n=1104 to 1904 step 40
5130 poke n,81:poke n+54272,8
5140 next
5150 for i=1905 to1906   :poke i,
     81:pokei+54272,8:next
5160 poke 1946,81:poke 1946+54272
     ,8
5170 for r=1986 to 2021
5180 poke r,81:poke r+54272,8:next
5190 poke 1981,81:poke 1981+54272
     ,8
5200 for i=1941 to1942   :poke i,
     81:pokei+54272,8:next
5210 for o=1943 to 1143 step-40
5220 poke o,81:poke o+54272,8:next
5230 for i=1142 to 1141 step-1
5240 poke i,81:poke i+54272,8:next
5250 poke 1101,81:poke 1101+54272
     ,8
5260 for l=1061 to 1026 step-1
5270 poke l,81:poke l+54272,8:next
5280 for i=1066 to 1106 step-40
5290 poke i,81:poke i+54272,8:next
5295 poke 1106,81:poke 1106+54272
     ,8
5300 poke 1105,81:poke 1105+54272
     ,8
5310 print"{17xneer}"
5320 n$="een programma van "
5330 print"{oranje}{4xrechts}";
5340 for i=1to18
5350 printmid$(n$,i,1);
5360 for j=1to40:next
5370 next
5380 n$="Klaster-soft"
5390 for i=1to12
5400 printmid$(n$,i,1);
5410 forj=1to40:next:next
5420 for i=1to2000:next
5430 data 1306,1346,1386,1426,146
     6
5431 data 1507,1467,1428,1469,150
     9
5432 data 1470,1430,1390,1350,131
     0
5433 data 1352,1392,1432,1472,151
     3
5434 data 1514,1475,1435,1395,135
     5
5435 data 1314,1313,1357,1397,143
     7
5436 data 1477,1518,1519,1480,144
     0
5437 data 1400,1360,1319,1318,132
     2
5438 data 1362,1402,1442,1482,152
     2
5439 data 1443,1484,1525,1444,144
     5
5440 data 1405,1365,1324,1323,132
     7
5441 data 1367,1407,1447,1487,152
     7
5442 data 1528,1529,1490,1450,141
     0
5443 data 1370,1329,1328,1332,137
     2
5444 data 1412,1452,1492,1532,153
     3
5445 data 1534,1535,1453,1454,133
     3
5446 data 1334,1335,1337,1377,141
     7
5448 data 1457,1497,1537,1378,141
     8
5449 data 1459,1499,1540,1500,146
     0
5450 data 1420,1380,1340
5499 return
5500 rem ********************
5501 rem *** maak tabel   ***
5502 rem ********************
5505 dimwo$(50)
5580 rem *** vullen elementen ***
5600 for a=1to50
5610 read wo$(a)
5620 next a
5999 return
6000 rem ************************
     **
6001 rem *** introduktie scherm *
     **
6002 rem ************************
     **
6010 printchr$(147);"{neer}hallo,"
6020 print:print
6030 print"dit is 'n programma om
      woorden te leren"
```

```
6040 print"schrijven. "
6050 print:print"je krijgt een wo
     ord te zien. "
6070 print"bekijk het goed,het ve
     rwdijnt zo weer. "
6090 print:print"daarna moet je h
     etzelfde woord intypen. "
6100 print"als 't woord klaar is:
     op return drukken"
6110 print"het programma kijkt of
      je het goed hebt"
6120 print"gedaan en houdt je sco
     re bij. "
6140 print:print"nu kun je bepale
     n hoelang het woord"
6150 print"moet blijven staan. "
6160 print"je kunt ingeven 1,2,3,
     4 of 5"
6170 print"1 betekent het kortst,
     5 het langst. "
6180 print"type nu het cijfer in. "
     :ti$="000000":f3=0
6182 geta$:ifti$>"000100"then f3=
     1:return
6183 if a$="" then 6182
6184 a=val(a$)
6186 if a<1 ora>5thenprint" 1t/m5
      please":goto 6182
6190 print"return. "
6200 print"{CLR-HOME}{3xneer}"
6499 return
6500 rem ************************
     *
6502 rem *** bepaal duur/woord **
     *
6504 rem ************************
     *
6540 wg=int(rnd(ti)*50)+1
6999 return
7000 rem *****************
7002 rem *** toon woord ***
7004 rem *****************
7010 print"{14xrechts}{RVS-aan}";
     wo$(wg);:print"{RVS-uit}"
7020 for i=1to700*a:next
7030 bl$=""
7040 l=len(wo$(wg))
7050 for i=1tol:bl$=bl$+" ":next
7060 print"{op}{14xrechts}"bl$
7499 return
7500 rem ********************
7502 rem *** input routine ***
7504 rem ********************
7570 print"het woord was"
7580 input" druk daarna op return"
     ;aw$
7999 return
8000 rem ************************
     ******
8002 rem *** beoordeel het antwoo
     rd ***
8004 rem ************************
     ******
```

```
8009 so=so+1
8010 if aw$=wo$(wg) then 8050
8020 fo=fo+1:tf=tf+1
8030 if fo<5  then goto 8040
8035 print"{neer}{10xrechts}dat w
     as vijf keer fout!"
8036 print"{neer}het woord was {RVS-a
     an}";wo$(wg);:print"{RVS-uit}"

8037 for i=1to2000:next:fo=0:goto
      8060
8040 f1=1:print"{neer}fout, doe h
     et nog maar eens!":goto 8499

8050 print"{CLR-HOME}{10xrechts}{5xne
     er}prima gedaan!"
8060 print"{10xrechts}{4xneer}wil
      je nog eens? (j/n)"
8070 geta$:if a$="" then 8070
8080 if a$="j"then  f2=1:print"{CLR-H
     OME}":goto 8499
8090 if a$="n" then 8499
8100 goto 8070
8499 return
8500 rem ******************
8502 rem *** eindroutine ***
8504 rem ******************
8507 if f3=1 and so=0 then goto 1
     00
8508 f3=0
8510 print"{CLR-HOME}{5xneer}je h
     ebt "so;" woorden getypt"
8520 print"{neer}hiervan waren er
      "tf;" fout en "so-tf;" goed"

8530 print"{neer}graag tot een vo
     lgende keer. "
8999 return
10000 data  receptie,spelling,verb
     eteren,spelfouten
10010 data glasservies,rijbewijs,j
     enever,groningen
10020 data vergadering,vakantiebun
     galow,centrum,busreis
10030 data deskundige,spandoek,opr
     oerpolitie,relletjes
10040 data gevangenis,bajes,kleute
     rschool,universiteit
10050 data rekenmachine,tekstverwe
     rker,inhoud,schijver
10060 data boekenwurm,spellingsfou
     ten,versnelling,bosatlas
10070 data docenten,onderwijs,inst
     antie,partij
10080 data vloerbedekking,computer
     spelletje,beoordelen,vooroor
     deel
10090 data verwennen,restaurant,di
     ner,vleugelpiano
10100 data bouwdoos.aardrijkskunde
     ,wiskunde,geschiedenis
10110 data verzekering,nieuwsdiens
     t,jurie,tentoonstelling
```

```
10120 data spionage,borgtocht

 *** EINDE LISTING ***


woorden
REGEL 10    165
REGEL 20    227
REGEL 30    194
REGEL 40    77
REGEL 50    154
REGEL 60    140
REGEL 75    252
REGEL 76    227
REGEL 77    0
REGEL 78    150
REGEL 79    93
REGEL 80    217
REGEL 81    252
REGEL 82    227
REGEL 83    105
REGEL 84    114
REGEL 85    227
REGEL 86    165
REGEL 100    40
REGEL 500    49
REGEL 1000    57
REGEL 1200    169
REGEL 1500    249
REGEL 2000    174
REGEL 2500    175
REGEL 3000    236
REGEL 3010    106
REGEL 3020    112
REGEL 3500    66
REGEL 4000    128
REGEL 5000    165
REGEL 5010    65
REGEL 5012    81
REGEL 5013    201
REGEL 5014    81
REGEL 5020    219
REGEL 5030    209
REGEL 5040    195
REGEL 5050    130
REGEL 5060    77
REGEL 5070    238
REGEL 5080    135
REGEL 5090    85
REGEL 5095    39
REGEL 5096    244
REGEL 5097    39
REGEL 5120    198
REGEL 5130    171
REGEL 5140    130
REGEL 5150    86
REGEL 5160    183
REGEL 5170    198
REGEL 5180    111
REGEL 5190    181
REGEL 5200    86
REGEL 5210    120
REGEL 5220    105
REGEL 5230    52
REGEL 5240    93
REGEL 5250    149
REGEL 5260    52
REGEL 5270    99
REGEL 5280    109
REGEL 5290    93
REGEL 5295    159
REGEL 5300    157
REGEL 5310    254
REGEL 5320    203
REGEL 5330    13
REGEL 5340    186
REGEL 5350    51
REGEL 5360    114
REGEL 5370    130
REGEL 5380    231
REGEL 5390    180
REGEL 5400    51
REGEL 5410    46
REGEL 5420    207
REGEL 5430    59
REGEL 5431    68
REGEL 5432    34
REGEL 5433    47
REGEL 5434    60
REGEL 5435    55
REGEL 5436    59
REGEL 5437    37
REGEL 5438    42
REGEL 5439    56
REGEL 5440    44
REGEL 5441    67
REGEL 5442    50
REGEL 5443    49
REGEL 5444    46
REGEL 5445    51
REGEL 5446    55
REGEL 5448    74
REGEL 5449    56
REGEL 5450    54
REGEL 5499    142
REGEL 5500    215
REGEL 5501    13
REGEL 5502    215
REGEL 5505    6
REGEL 5580    254
REGEL 5600    174
REGEL 5610    227
REGEL 5620    195
REGEL 5999    142
REGEL 6000    211
REGEL 6001    159
REGEL 6002    211
REGEL 6010    121
REGEL 6020    108
REGEL 6030    62
REGEL 6040    183
REGEL 6050    106
REGEL 6070    97
REGEL 6090    150
REGEL 6100    151
REGEL 6110    40
REGEL 6120    162
REGEL 6140    49
REGEL 6150    193
REGEL 6160    210
REGEL 6170    97
REGEL 6180    193
REGEL 6182    109
REGEL 6183    94
REGEL 6184    110
REGEL 6186    143
REGEL 6190    235
REGEL 6200    163
REGEL 6499    142
REGEL 6500    169
REGEL 6502    42
REGEL 6504    169
REGEL 6540    235
REGEL 6999    142
REGEL 7000    131
REGEL 7002    86
REGEL 7004    131
REGEL 7010    93
REGEL 7020    145
REGEL 7030    168
REGEL 7040    203
REGEL 7050    151
REGEL 7060    182
REGEL 7499    142
REGEL 7500    1
REGEL 7502    65
REGEL 7504    1
REGEL 7570    52
REGEL 7580    28
REGEL 7999    142
REGEL 8000    123
REGEL 8002    107
REGEL 8004    123
REGEL 8009    209
REGEL 8010    38
REGEL 8020    178
REGEL 8030    4
REGEL 8035    137
REGEL 8036    47
REGEL 8037    17
REGEL 8040    151
REGEL 8050    33
REGEL 8060    36
REGEL 8070    156
REGEL 8080    125
REGEL 8090    185
REGEL 8100    88
REGEL 8499    142
REGEL 8500    173
REGEL 8502    209
REGEL 8504    173
REGEL 8507    219
REGEL 8508    91
REGEL 8510    147
REGEL 8520    126
REGEL 8530    200
REGEL 8999    142
REGEL 10000    173
REGEL 10010    174
REGEL 10020    34
REGEL 10030    214
REGEL 10040    226
REGEL 10050    222
REGEL 10060    42
REGEL 10070    133
REGEL 10080    89
REGEL 10090    197
REGEL 10100    96
REGEL 10110    253
REGEL 10120    177

READY.
```

Vier op een rij.

Een overbekend spel wat ons door René Santing uit Wapse is toegezonden. Ik denk dat elke uitleg overbodig is. Probeer slimmer te zijn dan je tegenstander in dit geval de computer. Wie het eerst vier stenen op een rij heeft horizontaal, verticaal of diagonaal. Het speelbord bestaat uit 7 kolommen. Je hebt zelf de keus of je wilt beginnen of niet.

Met de grafische tekens in regel 19 wordt bedoeld: 1 maal shift +O, 26 maal Commodore toets +Y, 1 maal shift +P. In regel 20 shift +L, 26 maal Commodore toets +P, 1 maal shift +←.

```
 1 poke53281,3:poke646,0:dima(6
   ,7),b(7),c(7),d(2),w(6,2)
 2 data1,0,1,7,1,3,0,1,1,4,1,6,
   1,1,1,4,1,3,1,-1,4,7,1,3,999
   9,3,-6666,3,100,2
 3 data-50,2,3,1,-1,1
 4 goto62
 5 print"{CLR-HOME}*** vier op
   een rij ***{5xspatie}-spelre
   gels-":print
 6 print"het speelbord bestaat
   uit 7 kolommen.":print"u en
   ik gooien om de";
 7 print" beurt een schijf in";
   :print"een van de kolommen 0
    t/m 7."
 8 print"het is de bedoeling vi
   er eigen schijven":print"op
   een rij te";
 9 print" krijgen. dit geldt zo
   wel":print"horizontaal, vert
   icaal als diagonaal."
10 print"degene, die dit het ee
   rst lukt, heeft":print"gewon
   nen!":
11 print"{3xneer}de schijf{2xspatie
   }van{3xspatie}u{4xspatie}is:
   {10xspatie}' Q '";
12 print"de schijf{2xspatie}van
   {3xspatie}mij{2xspatie}is:{10xsp
   atie}' {WIT}Q{ZWART} '":print
   :return
13 input"{2xneer}wilt u beginne
   n{2xspatie}(ja/nee)";q$
14 q$=left$(q$,1):m=2
15 ifq$<>"j"then17
16 m=1
17 return
18 restore:fori=1to24:readx:next
   i:fori=1to6:b(i)=0:c(i)=0:read
   w(i,1),w(i,2)
19 forj=1to7:a(i,j)=0:nextj,i:z
   =0:a$="O77777777777777777777
   777777P"
20 f$="L///////////////////////
   ///:":return
21 input"in welke kolom zet u";
   k:if(k<=0)or(k>7)or(int(k)<>
   k)then21
22 ifc(k)>6then21
23 c(k)=c(k)+1:p=c(k):a(p,k)=9:
   z=z+1:return
24 print"{CLR-HOME}*** vier op
   een rij ***";tab(28)"zetnr.:"
   ;z:print"{4xneer}":ifm<2then
   26
25 print"{2xop}ik zet in kolom:"
   ;y;"{neer}"
26 printtab(6)a$
27 fori=6to1step-1:forj=1to7:k=
   (j-1)*4+6:printtab(k)"4";:b$
   ="{3xspatie}"
28 ifa(i,j)=0then31
29 b$=" {WIT}Q{ZWART} ":ifa(i,j
   )<>9then31
30 b$=" Q "
31 printb$;:nextj:print"{links}
   *":nexti:printtab(6)f$
32 print"{neer}kolom{3xspatie}1
   {3xspatie}2{3xspatie}3{3xspatie}
   4{3xspatie}5{3xspatie}6{3xspatie
   }7":print:printc$:return
33 ifh>2thenreturn
34 y=1:forj=2to7:ifb(y)=>b(j)then
   36
35 y=j
36 nextj:c(y)=c(y)+1:p=c(y):a(p
   ,y)=1:z=z+1:return
37 i=r:j=k:forn=1to4:ifa(i,j)<>
   0then41
38 ifc(j)+1<>ithen41
39 b(j)=b(j)+abs(w(g,1)):t=1:h=
   1:ifw(g,1)<>9999then41
40 c$="ik heb gewonnen, goed he
   !":n=4:h=2
41 i=i+v1:j=j+v2:nextn:return
42 forx=1to4:ifa(i,j)=0then46
43 q=1:ifa(i,j)<>9then45
44 q=2
45 s=s+1:d(q)=d(q)+1
46 i=i+v1:j=j+v2:nextx:ifs=0then
   54
47 h=3:c$="u heeft gewonnen. go
   ed hoor!"
48 ifd(2)=4then54
49 c$="":h=0:forg=1to6:p=1:ifw(
   g,1)>0then51
50 p=2
51 ifd(p)<>w(g,2)then53
52 gosub37:g=6
53 nextg
54 return
55 restore:fori=1to7:b(i)=0:next
   i:fore=1to4:readv1,v2,y1,y2,
   l1,l2:fork=y1toy2
56 forr=l1tol2:i=r:j=k:d(1)=0:d
   (2)=0:s=0:gosub42:ifh<2then5
   8
57 r=l2:k=y2:e=4
58 nextr:nextk:nexte:ift<>0then
   60
59 c$="gelijk spel!":h=3
60 return
61 y=int(rnd(0)*3+3):c(y)=c(y)+
   1:p=c(y):a(p,y)=1:return
```

```
62 gosub5:gosub18:gosub13:ifm<>
   2then64
63 gosub61
64 gosub24:gosub21:m=1:gosub24:
   m=2:gosub55:gosub33:ifh<2then
   64
65 gosub 24
66 fori=1to2500:next:print"{CLR-HOM
   E}";tab(210)"{ZWART}nog een
   Keer (ja/nee)";:inputnK$
67 nK$=left$(nK$,1):ifnK$="j"then
   62
68 print"{CLR-HOME}*+*+*+*+*+*+
   *+*+*+*+*+*+*+*+*+*+*+*+*+*"

69 print"+{37xspatie}+"
70 print"*{10xspatie}een produc
   tie van:{9xspatie}*"
71 print"+{37xspatie}+"
72 print"*{13xspatie}rene santi
   ng{12xspatie}*"
73 print"+{11xspatie}ten darper
   weg 46{10xspatie}+"
74 print"*{12xspatie}7983 Km{2xspat
   ie}wapse{11xspatie}*"
75 print"+{14xspatie}tot ziens!
   {13xspatie}+"
76 print"*{37xspatie}*"
77 print"+het programma verdwij
   nt binnen 10sec.+"
78 print"*+*+*+*+*+*+*+*+*+*+*+
   *+*+*+*+*+*+*+*+*"
79 forx=0to210:q=int(rnd(0)*16+
   .5):poKe53280,q:poKe53281,q+
   3:nextx:print"{HOME}{op}";
80 poKe53280,0:forx=1to1000:print
   "{RVS-aan}{ZWART} ";:nextx:for
   x=0to3000:next:sys64738
```

*** EINDE LISTING ***

vier op een rij

REGEL 1	249
REGEL 2	116
REGEL 3	22
REGEL 4	241
REGEL 5	33
REGEL 6	207
REGEL 7	228
REGEL 8	171
REGEL 9	174
REGEL 10	115
REGEL 11	186
REGEL 12	7
REGEL 13	89
REGEL 14	125
REGEL 15	1
REGEL 16	48
REGEL 17	142
REGEL 18	215
REGEL 19	177
REGEL 20	116
REGEL 21	93
REGEL 22	91
REGEL 23	39
REGEL 24	164
REGEL 25	253
REGEL 26	0
REGEL 27	240
REGEL 28	201
REGEL 29	128
REGEL 30	45
REGEL 31	226
REGEL 32	177
REGEL 33	235
REGEL 34	1
REGEL 35	85
REGEL 36	93
REGEL 37	72
REGEL 38	253
REGEL 39	164
REGEL 40	238
REGEL 41	249
REGEL 42	157
REGEL 43	247
REGEL 44	53
REGEL 45	198
REGEL 46	69
REGEL 47	140
REGEL 48	72
REGEL 49	236
REGEL 50	52
REGEL 51	48
REGEL 52	96
REGEL 53	201
REGEL 54	142
REGEL 55	51
REGEL 56	147
REGEL 57	169
REGEL 58	150
REGEL 59	207
REGEL 60	142
REGEL 61	31
REGEL 62	214
REGEL 63	244
REGEL 64	128
REGEL 65	243
REGEL 66	105
REGEL 67	115
REGEL 68	233
REGEL 69	51
REGEL 70	215
REGEL 71	51
REGEL 72	111
REGEL 73	37
REGEL 74	36
REGEL 75	212
REGEL 76	49
REGEL 77	155
REGEL 78	86
REGEL 79	25
REGEL 80	84

READY.

Mastermind C-16

Door Niek Meijer uit Maasland is ingezonden het overbekende spel **Mastermind**. De bedoeling van dit spel is om een getallen reeks te raden in zo min mogelijk beurten. Veel succes.

```
 2 gosub70
 4 :
 6 rem random kiezen
 8 fori=1to4:op$(i)=right$(str$
   (int(6*rnd(ti)+1)),1):next
10 fori=1to4:opl$=opl$+op$(i)+"
   ":next
12 :
14 rem invoeren 1-6
16 k=0
18 fori=1to4:p4$(i)=op$(i):next

20 k=k+2:r=1:fori=1to4:ik$(i)="
   ":next
22 do:rr=r:char,8+2*r,k,"{RVS-aan}
   {GEEL}"+ik$(r)+"{RVS-uit}"
24 getkeyy$
26 ify$=chr$(13)thenexit
28 ify$="{rechts}"thenr=r+1:goto
   36
30 ify$="{links}"thenr=r-1:goto
   36
32 ify$<"1"ory$>"6"then24
34 ik$(r)=y$:r=r+1
36 ifr>4thenr=1:elseifr<1thenr=
   4
38 char,8+2*rr,k,ik$(rr)
40 loop:char,8+2*rr,k,ik$(rr)
42 :
44 rem vergelijken
46 r$="n{links}"
48 fori=1to4:ifik$(i)=p4$(i)then
   r$=r$+"{ZWART}Q":ik$(i)="p":
   p4$(i)="P"
50 next
52 fori=1to4:forj=1to4:ifik$(i)
   =p4$(j)thenr$=r$+"{WIT}W":ik
   $(i)="p":p4$(j)="P"
54 nextj:nexti
56 char,19,k,r$+"{GEEL}"
58 ifr$<>"n{links}{ZWART}Q{ZWART}
   Q{ZWART}Q{ZWART}Q"andk<20then
   18
60 :
62 rem goede oplossing
64 char,10,22,"{RVS-aan}"+opl$
   +"{RVS-uit}"
66 getkeyy$:run
68 :
70 rem scherm
72 color0,10,3:color4,10,3
74 print"{CLR-HOME}{neer}{8xspatie
   }{ZWART}0@@@@@@@@@2@@@@.
76 fori=1to10:print"{6xspatie}{oran
   je}"right$(str$(i),2)"{ZWART}
   ]{9xspatie}]{4xspatie}]
78 print"{8xspatie}]@@@@@@@@@]{4xsp
   atie}]":next
80 print"{op}{8xspatie}+@@@@@@@
   @@[@@@@="

82 print"{8xspatie}]{GEEL}{RVS-aan}
   {4xspatie}?{4xspatie}{RVS-uit}
   {ZWART}]"
84 print"{8xspatie}-@@@@@@@@@="

86 char,10,0,"{oranje}a b c d{9xspa
   tie}mastermind":char,26,1,"E
   EEEEEEEEE"
88 char,25,4,"de oplossing":char
   ,25,6,"bestaat uit"
90 char,25,8,"vier getallen":char
   ,25,10,"tussen 1 en 6."
92 char,25,13,"succes!":return

*** EINDE LISTING ***
```

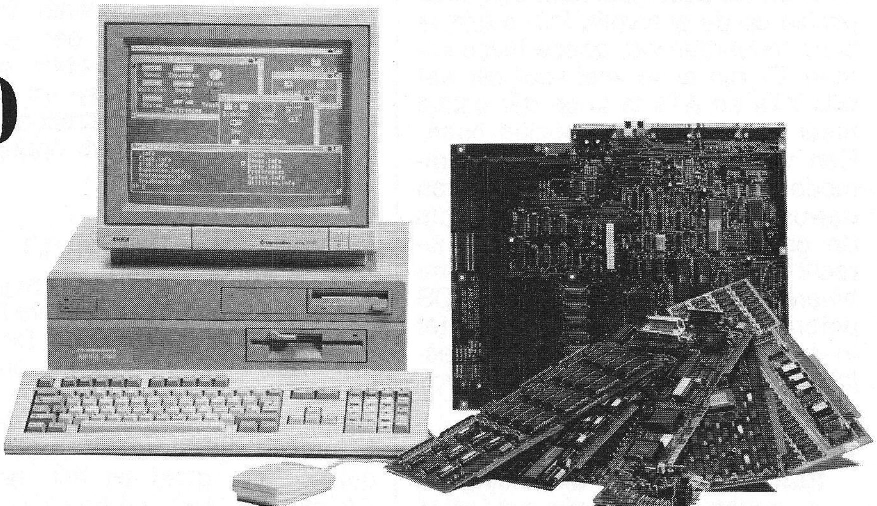
REGEL 2	244	REGEL 52	117
REGEL 4	58	REGEL 54	209
REGEL 6	22	REGEL 56	27
REGEL 8	130	REGEL 58	55
REGEL 10	64	REGEL 60	58
REGEL 12	58	REGEL 62	177
REGEL 14	137	REGEL 64	190
REGEL 16	45	REGEL 66	219
REGEL 18	204	REGEL 68	58
REGEL 20	144	REGEL 70	81
REGEL 22	65	REGEL 72	68
REGEL 24	23	REGEL 74	77
REGEL 26	202	REGEL 76	105
REGEL 28	31	REGEL 78	240
REGEL 30	160	REGEL 80	113
REGEL 32	149	REGEL 82	168
REGEL 34	245	REGEL 84	7
REGEL 36	77	REGEL 86	229
REGEL 38	192	REGEL 88	255
REGEL 40	230	REGEL 90	125
REGEL 42	58	REGEL 92	34
REGEL 44	197		
REGEL 46	87		
REGEL 48	56	READY.	
REGEL 50	130		

vervolg van pag. 36

```
885 if at<5 then 825
890 print:print"Je bent te zwaar...."
895 print"Je moet iets weggooien."
900 k=3:goto 850
905 :
910 rem attributen controle
915 for i=1 to 7
920 if me$(i)<>m$(i) then return
925 next i:e=1:return
930 :
1000 rem gegevens in dataregels
1005 :
1010 rem info schatkaart
1015 data"Het is een oude kaart van een eiland.
1020 data"Het lijkt wel of er iets op aangegeven
1025 data"staat. Een kruis, of zo.
1030 data"Ligt daar de schat van Rooie Pier ?
1035 data 2,2,2,2
1040 rem info sleutel
1045 data"Het is een grote roestige sleutel.
1050 data"Zeker in jaren niet gebruikt.
1055 data"Er zitten resten zeeklei op.
1060 data"Ook nog een jaartal: 1678
1065 data 4,4,4,4
1070 rem info schep
1075 data"Het is een gewone stalen schep.
1080 data"Er is niets bijzonders aan te zien.
1085 data"De schep is zo te zien nog nieuw.
1090 data"Die kan dus wel van pas komen.
1095 rem info boot
1100 data"Het lijkt een wrak bootje.
1105 data"Niet erg zeewaardig in elk geval.
1110 data"Ik zou hiermee niet de zee op durven!
1115 data"Roeispanen zijn er ook niet bij."
1120 data 7,7,7,7
1125 rem info zwaard
1130 data "Zo te zien is het zwaard kort geleden
1135 data"nog gebruikt. Er zitten sporen van
1140 data"menselijk bloed aan.
1145 data"Erg zwaar is het ook!
1150 :
1155 rem titel
1160 data"****************************************
1165 data"      De Schat op het Piraten Eiland
1170 :
1175 rem inhoud lokaties
1180 rem lokatie 1
1185 data"Je bent in een oude vissers-hut.
1190 data"In de verte ruist de zee.
1195 data"In een hoek staat een scheepskist.
1200 rem lokatie 2
1205 data"Je staat in de duinen.
1210 data"Achter je is een oude hut.
1215 data"In het oosten zie je een oud kasteel
1220 rem lokatie 3
1225 data"Je staat nu op het strand.
1230 data"Rechts staat een oude hut.
1235 data"Achter de branding ligt een eiland.
1240 rem lokatie 4
1245 data"Het eiland waarop je bent, ziet er
1250 data"geheimzinnig uit. Er zijn zo te zien
1255 data"geen bewoners. Er staat een totempaal.
1260 rem lokatie 5
1265 data"Het is aardedonker in deze grot.
1270 data"Geen erg prettige omgeving. Ik zou
1275 data"maar snel vertrekken, als ik jou was!
1280 :
1285 rem lokatie 6
1290 data"Je krijgt natte voeten in dit moeras.
1295 data"Er zijn ook nogal wat muggen hier.
1300 data"Ze steken ook nog... au.. au..WEGWEZEN!
1305 rem lokatie 7
1310 data"Deze oude puinhoop was vroeger een
1315 data"kasteel. Er is niet veel van over.
1320 data"Alleen een van de torens staat er nog.
1325 rem lokatie 8
1330 data"Dit oerwoud is niet toegankelijk voor
1335 data"gewone avonturiers. Je hebt niet eens
1340 data"een geweer bij je!
1345 :
1350 rem instructies
1355 data noord,zuid,west,oost
1360 data mee,help,kijk,pak,weg
1365 :
1370 rem relatie-tabel
1375 data 2,5,3,6,0,1,0,7,0,4,0,1
1380 data 3,0,0,0,1,0,0,8,7,8,1,0
1385 data 0,6,2,0,6,0,5,0
1390 :
1395 rem attributen
1400 data schatkaart,,sleutel
1405 data ,schep,boot,,zwaard
1410 :
1415 rem bezetting
1420 data 1,0,1,0,1,1,0,1
1425 :
1430 rem hulp
1435 data "Ga niet in die donkere grot!
1440 data "Zonder boot gaat het niet.
1445 data "Waar ligt de sleutel ?
1450 data "Bekijk de kaart nog eens goed.
1455 data "Ik kan je helaas niet verder helpen.
```

Het Amiga-model 1000 was eigenlijk een droommachine. De Droom van Jay Minor en zijn ontwerpteam was een 68000 computer te ontwerpen die alle andere PC-hardware op het gebied van graphics en sound zou verslaan. Zoals met alle dromen vervagen bij idealistische systeemontwerpen de grenzen van de realiteit. De Amiga 1000 is een aantrekkelijke machine voor de artiest en creatieve hobbyist, maar telt eigenlijk niet mee op de kille zakelijke MS-DOS-markt. Commodore denkt daar nu met de open systeem architectuur (OSA) van de Amiga 2000 verandering in te brengen.

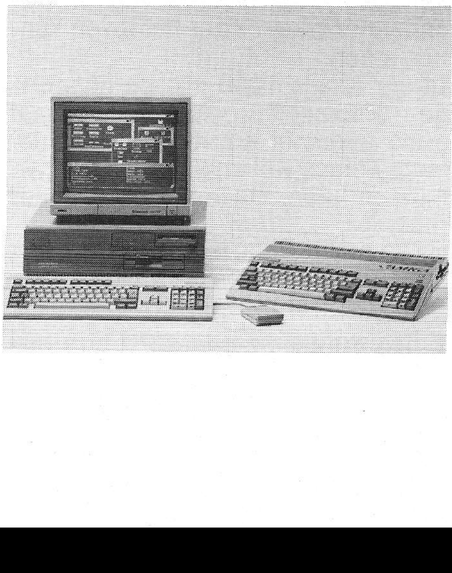
AMIGA 2000



Een open system architectuur maakt nog geen nieuwe computergeneratie

Wie op de computerbeurzen in de laatste maanden van 1986 een scherp oog voor prijzen had kon duidelijk in de Amiga-koffiedik kijken. De prijzen voor een compleet 1000-systeem met 512K daalden fluks van zo'n *f* 4500,- tot minder dan 3000 piek en zelfs grootwinkel-bedrijven toonden de Amiga tegen lokprijzen in hun etalages. Commodore hield duidelijk uitverkoop. Even bekroop de liefhebber de vrees dat het uit was met CBM's 6800-avontuur, maar algauw deden mysterieuze persberichten over een mogelijke opvolger de ronde en vroeg Commodore de PC-pers voorlopig het zwijgen te bewaren. Primeurjagers moesten natuurlijk weer zonodig dit persembargo doorbreken en tal van "tests" werden al aan het geduldige papier toevertouwd. Wij keken echter eerst de kat uit de boom om u een gefundeerde mening te kunnen geven.

Bij de introductie werd de Amiga door menig computerjournalist als een hachelijk avontuur gezien. De hamvraag was of de fraaie graphics, animaties en brute 68000 kracht het wel zouden kunnen bolwerken tegen het almachtig regerende MS-DOS PC-(kloon)-imperium. Een ander veel gehoord bezwaar tegen de Amiga 1000 was de onvoldoende en trage hardware-ondersteuning. Die ene ingebouwde 3.5 inch 720K- drive zette qua snelheid en mogelijkheden weinig zoden aan de dijk. Bovendien waren er door de gesloten systeem-architectuur slechts weinig uitbreidingsmogelijkheden. Een enkeling sprak daarom zelfs van een generatiekloof tussen de 68000 en ondersteunende Commodore hardware.



Niettemin wist de Amiga binnen korte tijd een enthousiaste schare gebruikers te verwerven en daar droeg de geleidelijk dalende systeemprijs flink aan bij. Deze gebruikers kwamen vooral uit de artistieke en audiovisuele productiehoek. Een heel aparte zakelijke en hobby-kategorie. Langzaam kreeg de Amiga-fan met brede beurs ook steun van onafhankelijke hardware-fabrikanten. Harddisks, RAM-uitbreidingskaarten, digitizers en extra floppystations kwamen mondjesmaat tegen forse bedragen beschikbaar. Bovendien gingen een aantal belangrijke softwarehuizen geheel op de Amiga-toer en brachten imposante business pakketten uit.
Commodore vond deze horizontale markt echter niet genoeg en mikte vertikaal met de MS-DOS compatibele **Transformer** en later de **Sidecar**-emulator. Al bij de introductie van Sidecar bleek deze Intel 8088 CPU-emulator een overgangsprodukt. Meer dan een wat povere bijwagen

voor MS-DOS programmatuur stelde Sidecar niet voor. Er zat echter nog meer in het vat zoals op de onlangs gehouden CEBIT bleek. Commodore's nieuwste zakelijke 68000-troef is de Amiga 2000 met een OSA-ontwerp moet de MS-DOS markt voor de Amiga openleggen.

De troeven

Om een serieuze gooi naar een hoge positie op de overvolle PC- markt te doen moet men van goede huize komen. Er zijn al zo veel voor elk wat wils XTs en ATs te koop dat eentje meer of minder geen opzien baart. Een voldongen feit waar de Commodore- ontwerpers zich terdege van bewust waren. Zij besloten daarom de grafische, muzikale en rekenkracht van de Amiga 1000 te combineren met het enorme MS-DOS potentieel door een hybride-computer in een huis te ontwerpen. Het resultaat is de Amiga 2000 met de volgende sterke troeven:

- Een **Open Systeem Architectuur** (OSA) maakt uitbreiding via de zeven ingebouwde expansion slots mogelijk. Ook zijn er nog twee extra slots voor een 68020 CPU met memory management en een floating-point chip, en een nieuwe PAL composiet-videokaart.
- Een**XT-** (Janus Card) en een **AT-** -kaart die afzonderlijk of onder multitasking (Amiga DOS of UNIX 5.2) in de Amiga 2000 kan draaien. Via dubbele slotconnectoren maakt de kaart aan de ene kant kontakt met de Amiga- en aan de andere kant met de IBM PC hardware (de ingebouwde "Sidecar).
- Het gebruik van **Hard Cards/disks** die zowel Amiga- als MS-DOS- partitions kunnen bevatten. M.a.w. op een en dezelfde on board harde schijf kan het disk operating system zowel Amiga- als MS-DOS-files wegschrijven en lezen.
- Het speciale **video-slot** maakt in de toekomst HIRES (640 x 512) non-interlaced composiet video mogelijk. Het einde van al dat gehannes met losse video-kaarten, genlockers e.d. is dus nabij.
- Er is nu ruimte voor drie, twee 3.5 inch en een 5.25 inch, floppydrives in de systeemkast.
- **Kickstart** zit bij de 2000 in ROM hetgeen het booten aanzienlijk versnelt.
- Standaard beschikt de Amiga 2000 al over 1 Megabyte aan RAM. Maximaal kan 8 MB aan vrij RAM geadresseerd worden.
- Een on board **klokkalender** met batterij-voeding.

Al deze extra hardware maakt de Amiga 2000 tot een waar **Power House**, maar niet tot een geheel nieuwe machine. De 7.14 MHz 68000 CPU en de DMA, display- en geluidchips bleven immers hetzelfde. Beter is het om van een sterk opgepepte 1000-versie te spreken.

De Amiga 2000

Commodore's zakelijke Amiga-telg ziet er veel meer als een echte PC uit dan de oude 1000-versie. De systeemkast is nu van metaal en flink hoger en dieper geworden. De 1081 kleurenmonitor lijkt nu niet meer op een soort "waterhoofd" maar past qua grootte goed bij het metalen Amiga-huis. Het verhoogde voorfront biedt plaats aan twee 3.5 inch en een 5.25 inch drive. Verder verhuisden de muis- en de joystick-aansluitingen naar de voorzijde.

De achterzijde van de 2000-kast blijkt een ware verademing. Weg zijn de afwijkende Commodore parrallelle en seri*f*le connectoren met het voortdurend dreigende gevaar van een noodlottige kortsluiting bij het gebruik van een verkeerd kabeltje. Dat is nu allemaal PC-standaard geworden. Ook zitten er aan de achterzijde nog een RGB video-poort, een aansluiting voor een externe Amiga drive en twee RCA jackpluggen voor het stereo-geluid. De Composiet video-jackplug is verdwenen.

Er zijn twee keyboard-uitvoeringen leverbaar: Een Amerikaanse versie met 94 toetsen en een Internationale versie met 96 toetsen. Nieuw zijn de HELP-, de ENTER-, en de numerieke +,-, *, /-toetsen. De cursortoetsen zijn in een T- inplaats van een kruispatroon gerangschikt. Een ruimere layout maakt het nieuwe keyboard veel overzichterlijker dan de krappere oude uitvoering.

9 Sleuven

Na het open van de systeemkast worden de negen (er zijn zeven uitbreidingsslots, maar twee daarvan vormen via dubbele connectoren een **brug** tussen de Amiga- en PC-bus!) connector-slots zichtbaar. Een zee aan uitbreidingsmogelijkheden in vergelijking met de enkelvoudige externe connector van de 1000. Vijf van de slots gebruiken de Amiga-systeembus en vier slots gebruiken een speciale secundaire IBM PC XT-compatibele bus. Middels twee 36-pins connectoren is ook al op de aangekondigde AT-kaart gerekend.

Met de zeven uitbreidingsslots zijn diverse configuraties mogelijk. Een aantrekkelijke optie is het gebruik van de **A2088 XT** PC-emulator (en later eventueel de AT-emulator) in een van de brugslots. Dan blijven er nog drie slots voor PC-kaarten, bijvoorbeeld een EGA of modem en een hard card en drie voor Amiga-kaarten, bijvoorbeeld een RAM-kaart of harddisk-controller over. Wie nog een extra Amiga kaartsleuf wil schuift de PC-kaart gewoon een slot naar links en houdt dan vier Amiga- en twee PC-slots over.

Naast de gewone uitbreidingsslots zijn er nog twee speciale Amiga-slots:

° Een **86-pins Amiga busslot** voor de 68000 CPU. Hierin kan bijvoorbeeld een 68020 mathematische co-processor met floating point en een eigen memory management gestoken worden.

° Een speciaal **PAL Composiet Video Board-slot**.

Gebus

Twee processoren op een systeemkussen slaapt daar wellicht de bugduivel tussen? Op het eerste gezicht een beangstigende vraag voor een hybride machine als de Amiga 2000 waarin immers twee CPU's onder multitasking kunnen draaien. Toch zijn er uit de PC- praktijk tal van voorbeelden bekend waarbij meerdere processoren broederlijk naast elkaar de systeembus kunnen nemen. Veelal gebruiken de ontwerpen dan de truuk om een van de CPU's tijdelijk in slaap te sussen terwijl de andere aan het rekenen slaat. Dat gaat dikwijls goed, maar toch is onder andere bij Turbo-boards gebleken dat twee processoren in één computer elkaar naar het leven kunnen staan.

Bij de Amiga 2000 zijn de onderlinge afspraken gelukkig allemaal goed geregeld waarbij de constructeurs helaas wel enige veren hebben moeten laten. Daarover echter later meer.

Bij de Amiga 2000 is sprake van maar liefst drie bussen. Deze kleine busonderneming valt uiteen in:

- De **68000 32-bits CPU-bus** met

86 adreslijnen. Het gaat duidelijk om een soort snelbus, want er zit geen enkele buffer tussen het CPU-slot en de Motorola 68000 microprocessor.

- De **Amiga-bus** met 100 adreslijnen voor CPU-, externe interrupt- en bus-verkeersregelsignalen. Alle lijnen zijn gebufferd wat het construeren van de interfaces voor de uitbreidingskaarten aanzienlijk vergemakkelijkt. Jammergenoeg is deze busstructuur niet compatibel met de oude Zorro buskaart voor de Amiga 1000.
- De **PC-bus** is naar keuze de bekende 62-lijns XT-bus of de 16-bits AT-bus.

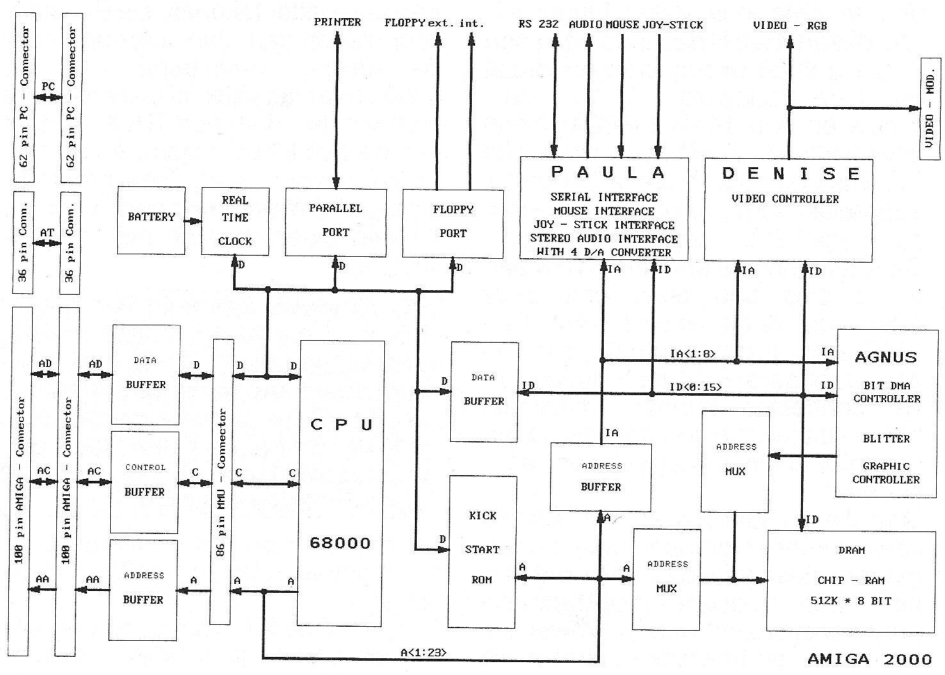
Om al het busverkeer in goede banen te leiden heeft de Amiga een eigen perron-opzichter in de vorm van **AutoConfig**. Het AutoConfig protocol versie 1.2 zorgt er voor dat alle kaarten probleemloos op de bus stappen. Hiervoor zitten op de Amiga-bus de lijnen CONFIG-IN en CONFIG-OUT. Bij het opstarten of na een RESET staan alle uitbreidingskaarten als het ware op de bus te wachten, de ongeconfigureerde status. CONFIG-IN checkt de eigenschappen (signature data) van de eerste kaart en geeft deze aan de CPU door. Vervolgens configureert dan de Kickstart software de kaart aan de hand van de opgegeven adresruimte, initialisatiecode en geheugengebruik. Is het board aldus geconfigureerd dan zendt het een CONFIG-OUT-signaal uit en begint het proces met de daaropvolgende kaart weer van voren af aan. Bij de 2000 verloopt AutoConfig geheel automatisch via het 256K meten on board Kickstart ROM.

Het RAM

De Amiga 2000 is standaard uitgerust met 1MB on board RAM. Desgewenst kan de geheugen-capaciteit via expansion boards en AutoConfig tot 8MB worden uitgebreid.

De memory boards die de 68000-bus gebruiken bieden het zogenaamde **fast memory**. Fast wil zeggen dat het RAM de bus-access niet met de video Blitter-chip behoeft te delen. Dat was wel het geval op de Amiga 1000 waarbij de Blitter, dankzij zijn hogere systeem- prioriteit, schaamteloos buscycli stal van het interne **chip memory** hetgeen de programma-uitvoer vertraagde.

Op de Amiga 2000 bestaat de helft van het 1 MB grote RAM (dus 512 K) uit fast- en de andere helft uit chip memory. De Blitter kan daarbij rustig te keer kan met het chip videogeheugen terwijl de software aanzienlijk sneller in het fast Ram draait. 2MB RAM- uitbreidingskaarten kosten circa 1100 gulden.



Een specificieke Amiga 2000-toepassing is het gebruik van **dual- port-Ram**. Dit 128 KB omvattende RAM-geheugen buffert het dataverkeer tussen het 68000- en PC-deel van de Amiga. Op dit RAM hebben zowel de Motorola 68000 CPU als Intel 8088/80286 compatibele microprocessoren van de Janus/AT-kaart toegang. Beide processoren maken echter gebruik van een verschillend dataformaat. De 68000 gebruikt het **IFF**- en de PC-processor het **ASCII**- formaat. Via een ingenieus accessysteem zijn de Amiga-ontwerpers er echter ingeslaagd om de verschillende data-formaten als het ware naar weerszijden om te spiegelen tot het gebruikte formaat. M.a.w. de Amiga leest de bits in een tegengestelde volgorde af als de PC-kaart doet.

De coprocessor-chips

Ook in de Amiga 2000 treft u het roemruchte Chiptrio uit het 1000-model aan:

* De Bit-Blitter **Agnus** zorgt voor de flitsende animaties met 60 beelden per seconde en kan daarbij verscheidene bewegingsbronnen, bijvoorbeeld los bewegende armen en benen, aan elkaar koppelen. Agnus controleert 25 DMA-kanalen, spoort sprite-botsingen op en ondesteunt de grafische mogelijkheden van Denise.

* De DMA-chip **Paula** beheerst de geheugentoegang van de coprocessoren. Daarnaast controleert Paule de I/O-besturing van de parallelle en seri*f*le interfaces, de control ports, het uitlezen van het toetsenbord en de audio-in/output. Het stereo- deel omvat vier toonkanalen met twee stereo-uitgangen, programmeerbare sampling-rate, amplitude, 9 octaven en frequentie-modulatie.

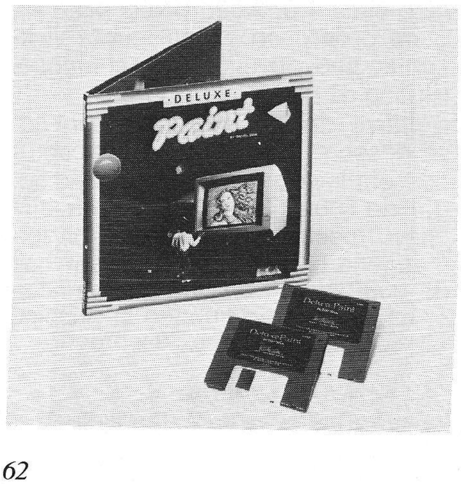
* De grafische chip **Denise** biedt een palet van 4096 kleuren, stuurt de sprites en verplaatst de "playfields". In samenwerking met Agnus kan Denise circa 1.000.000 pixels per seconde optekenen en gekleurde vlakken zo snel opvullen dat het menselijk oog dit niet merkt.

Behalve de on board systeem-coprocessors zijn ook add on coprocessors leverbaar. Voor de hand liggen de 68020 en 68030 mathematische processors met een ondersteunend cache memory, maar ook de 8088/80286 en hun mathematische processoren kunnen in principe als coprocessor gebruikt worden.

XT- en AT-kaarten

Het **A2088 board** of Janus-kaart is

een in feite een zelfstandige XT-hoofdkaart. Een Sidecar op een kaart dus. De 8088 microprocessor draait onder de standaard 4.77 Mhz klokcyclus en een 16KB EPROM bevat alle noodzakelijke BIOS-routines. Het BIOS is door Commodore behoorlijk aangepast om communicatie met de Amiga 68000 mogelijk te maken.
De A2088 heeft standaard 512 K aan boord plus nog eens een extra 128K voor het dual-port RAM. Een SMC 9268 floppy controller chip en een standaard 5.25 inch floppy drive connector kunnen maximaal twee (een interne en een externe drive) daisy chained floppystations aansturen.
Om alle XT-functies op een enkele kaart te kunnen proppen moest Commodore eigen (custom)-chips ontwerpen. Een speciale **multifunction chip** functioneert daarbij zowel als een DMA- en interrupt controller als een timer voor alle bussignalen. Andere chips "bedriegen" het Sidecar-deel van de Amiga 2000. De machine "gelooft" dat hij een echte XT is en via het dual-port memory worden alle toetsaanslagen en I/O-opdrachten van de Amiga als zijnde een PC XT geëmuleerd.
Verder biedt de XT-kaart een extra socket voor de 8087 mathematische coprocessor.
Aan de onderzijde van de A2088 PC emulator-kaart zitten twee grote connectoren. De grootste connector wordt in de Amiga bus- connectorslot gestoken. De kleinste in het PC-deel van de 2000.
De A2088 wordt natuurlijk ook softwarematig ondersteund vanuit Kickstart en de Intuition werkbank. Via interrupts trekt de ene computer de aandacht van de andere. Zo heeft de XT-kaart een eigen PC-ikoon (naar keuze monochroom of in kleur) op de werkbank waarin informatie en graphics opgetekend worden. Stel de A2088 wil een grafiek op de



Amiga-monitor tekenen. De PC-emulator zendt dan een interrupt naar de Amiga, vervolgens kijkt de 68000 naar het video display memory deel van het dual-port RAM en gaat dan via een EXEC-signaal tot de grafische uitvoer over. Het programmaatje **PCWindows** opent voor de XT een eigen venster met informatiescherm.

PC Windows kan ook, met maximaal 7 interrupts, onder multitasking draaien. Zo is het bijvoorbeeld mogelijk om dBASE op de XT te laten sorteren terwijl de Amiga zelf dan voor de tekstverwerking of het maken van animaties vrij blijft.

De prijs van de A2088 Janus-kaart zal ongeveer rond de *f* 1400,- komen te liggen.
Gelijk met de XT-kaart werd een AT-insteekkaart aangekondigd. Wij konden er nog geen hand op leggen, maar volgens de specificaties bevat dit emulator board:

° Een 80286 CPU met een kloksnelheid van 8MHz
° 512KB (plus 128 dual-port) aan vrij RAM
° Een 16 KB BIOS-ROM
° Een floppy-controller voor en 5.25 1.2 MB en een 3.5 inch 720 KB drive
° Een socket voor de 80287 mathematische coprocessor

De prijs van dit **A2286** AT-board wordt op iets van *f* 2200,- geraamd. Voor de toekomst denkt Commodore nog aan een 386 emulator, maar daarover staat niets vast.

Harddisk

Een van de bezwaren die aan de Amiga kleefden was de relatief hoge prijs (*f* 4000,-) van een harde schijf. Weliswaar kon de gebruiker voor zo'n 2000 gulden een Sidecar kopen en daar nog eens voor 1000 piek aan 20MB PC hard disk in zetten, maar het bleef behelpen en duurder dan bij de doorsnee IBM PC-kloon.
De Amiga 2000 heeft daar nu via de uitbreidingsslots verandering in aangebracht. Voor *f* 1800,- krijgt de gebruiker de beschikking over een 20 MB 3.5 inch harddisk met controller op kaart. Desgewenst is ook een losse harddisk controller (Type 2094 voor het ST-506 en SCSI-protocol) verkrijgbaar. Deze controller kan maximaal twee ST-506 of zeven (!) SCSI drives in elke willekeurige combinatie aansturen. Aan externe data-opslag dus geen gebrek.
De disk access van het Amiga-DOS creïrt twee verschillende partitions. Een deel van de harde schijf wordt gebruikt voor de Amiga-files en het overige deel krijgt via FDISK de beschikking over MS-DOS allocaties. Het PC-deel van de harddisk heet op de werkbank **JH0** en kan net als elke andere harde Amiga-schijf gebruikt worden. Speciale hardware drivers houden de verschillende partions als afzonderlijke hard disks uit elkaar. Jammer vinden wij dat booten van de harde schijf (nog?) niet mogelijk is. Het opstarten gaat nog steeds via de DF0 floppy drive.

Floppystations

De standaard-configuratie van de Amiga 2000 doet, wat de drives betreft, tamelijk kaal aan. Voor de *f* 3000,- die de 2000 exclusief monitor kost zit er slechts een 3.5 inch diskdrive in de systeemkast. Bij een PC-kloon zitten er voor een zelfde bedrag minimaal twee diskettestations in de kast, maar zo'n kloon kan uiteraard veel minder dan de Amiga 2000.
Toch zal de serieuze gebruiker nog wat dieper in de beurs moeten tasten, want met een floppystation blijft het traag werken. Er is keuze uit een intern 3.5 inch model (circa *f* 500,-, maximaal twee drives in de kast) en een 5.25 inch model (circa *f* 600,- , maximaal 1 station). Wie nog meer drives wil zal de externe connector moeten gebruiken. Dat zelfde geldt voor bezitter van de externe drives voor de Amiga 1000.

Video display

De Amiga is beroemd om de kwaliteit van zijn 640 x 512 colour graphics en videoanimaties. Toch was er sprake van enige kritiek. Videoproducenten waren niet tevreden over de kwaliteiten van de genlockers en het PAL-beeld. In de praktijk was synchroniseren niet zo simpel als Commodore het deed voorkomen en ook de professionele videokwaliteit viel tegen.
De Amiga 2000 kent minder synchronisatie-problemen. Bovendien heeft Commodore een speciale PAL-insteekkaart ontworpen waarvan de beeldkwaliteit overeenkomt met hetgeen men in videostudio's gewend is.
N.B.: De Amiga 2000 genereert, integenstelling tot model 1000, zonder video-kaart **geen com-**

posiet videobeeld. Videogebruikers moeten er dus de PAL-kaart bijkopen.
Verder komt er op korte termijn een speciale CAD-monitor voor het creëren van hoogwaardige graphics en het maken van kwaliteits-dia's.
Met de Sidecar vormde EGA-compatibele software een probleem. De EGA-kaart verschilt de veel van de Amiga graphics-hardware om een goed kleurenbeeld te kunnen krijgen. In een van de PC-slots kan de gebruiker EGA compatibele MS-DOS software nu gewoon een EGA/PGA-kaart duwen en de Amiga 2000 genereert prachtige kleuren beelden op een EGA-compatibele monitor.

Geluid

Alle geluidsspecificaties zijn, in vergelijking met de Amiga 2000, vrijwel hetzelfde gebleven. 4 Gescheiden geluidskanalen geconfigureerd als twee maal twee stereokanalen, 4 ingebouwde D/A-converters en MIDI-compatibiliteit maken de Amiga 2000 tot een komplete muziekstudio. De geluids-input kan zowel via microfoon als andere stereo-apparatuur en MIDI-compatibele muziekinstrumenten. Voor een optimale weergave raden wij het gebruik van een separate stereoinstallatie aan. De kwaliteit van de ingebouwde luidspreker is maar zo zo. Verder bleef de uitspraak van gedigitaliseerde stemmen zeer Amerikaans.
Van de PC-geluids weergave over de Amiga 2000 stereokanalen moet u zich geen illusies maken. Die is ronduit slecht, maar dat kan gezien het PC-ontwerp niet anders. Een PC-kloon kan ook niet meer dan enkele piep- of bliep-geluiden voortbrengen.

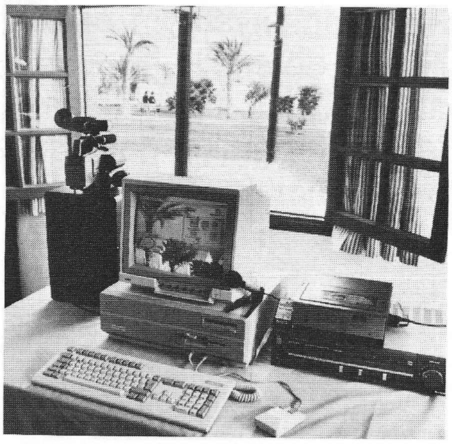
Tegenvallers

De Commodore-ontwerpers hebben met de Amiga 2000 een goede prestatie geleverd. Het is echter niet alles goud wat aan het 68000-firmament blinkt. Wij hebben toch nog wel wat puntjes van kritiek waarvan hieronder een korte samenvatting.

- De Amiga 2000 blijft een **bouwdoos**. Met de basisconfiguratie is de zakelijke gebruiker of serieuze hobbyist er nog lang niet. Een volledig aangeklede (AT-kaart, harddisk, video-adapter) Amiga kost bijna het driedubbele van de basisconfiguratie.
- Zonder coprocessor of nieuwe 68000-telg (waarom de oude 16-bits busser ingebouwd?) is de 2000 eigenlijk niet meer dan een upgrade van de Amiga 1000.
- Problemen met de PC-videodisplay. De Amiga 2000 biedt de keuze uit de volgende PC-videomodes: monochroom (de kleur is naar keuze uit het Amiga-pallet), 320 x 200 pixels vier-kleuren en 640 x 200 twee-kleuren display. Voor een Hercules-, EGA- of PGA-display zijn een aparte PC-kaart en compatibele monitor nodig. Andere bezwaren betreffen het ontbreken van flashing characters en een niet exacte video display op het Amiga-scherm. Dat laatste is met name storend bij het scrollen (de tekst springt meer dan zij scrolt) en lelijke animatie-overgangen bij o.a. Flight Simulator.
- Tot op heden is geen PC-muis-emulatie beschikbaar. De Amiga-muis werkt alleen met Amiga-programmatuur. Commodore zegt aan een MS Mouse-emulatie te werken.
- Op het moment van schrijven was alleen de parallelle printerpoort, via het driverprogrammaatje LPT1, voor de PC-boards beschikbaar. Aan de seriële aansturing wordt nog gewerkt.
- Kickstart versie 1.2 in ROM bleek in een aantal gevallen incompatibel met oudere, onder versie 1.1 draaiende, games of software. Pagesetter bijvoorbeel verwacht 1.2. Als gebruiker ben je dan verplicht om onder Kickstart 1.1 resp. 1.2 op te starten en dat geeft weer incompatibiliteits-problemen met de nieuwe ingebouwde hardware.

De doelgroepen

De nieuwe Amiga 2000 probeert van meerdere walletjes te eten. Iedereen die een Amiga met hard disk en maximaal 8 MB aan vrij (en Fast) Ram wil zal zich in deze machine kunnen vinden. AV-producenten worden met de betere videokwaliteit op hun wenken bediend.
Een geheel andere markt vormen de serieuze hobbyisten en veel eisende zakelijke gebruikers. Stoeien met een Motorola 68000 of 68020, sublieme business graphics, multitasking in combinatie met een AT- of zelfs een 386-kaart en draaien onder UNIX 5.2 met Memory Management Unit (MMU) zijn voor deze doel-



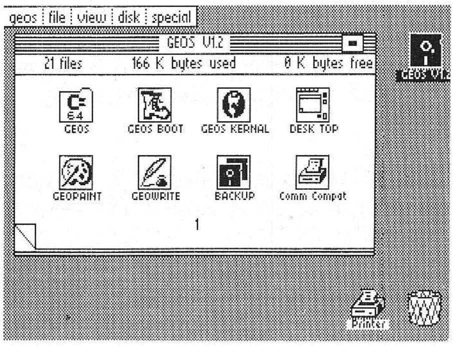
groep aantrekkelijke mogelijkheden. Hen verwezenlijken zal echter niet goedkoop zijn, maar voor een gelijkwaardige PC-kloon moet immers ook een berg geld worden neergeteld.
Als laatste doelgroep noemen wij de wetenschappers en educatieve instellingen die de Amiga als experimentele, respectievelijk leercomputer kunnen inzetten.

Met de Amiga 2000 heeft Commodore een unieke hybride PC op de markt gebracht. Voor de integratie van een 68000 met een MS-DOS machine en een OSA valt veel te zeggen en niet iedereen zal even zwaar aan de genoemde bezwaren tillen. De eerste stap naar een geheel nieuwe computergeneratie is gezet.
Blijft over de onaffe indruk die de Amiga op ons maakte. Alle mogelijkheden zijn nog niet benut en wij verwachten op niet al te lange termijn een echte nieuwe Amiga met een krachtiger 68000-telg en minder compatibiliteits-problemen.
Als afsluitende opmerking iets over de software-piraterij rond de Amiga. Misschien valt het u op, dat er zo weinig Amiga software te koop is. Helaas heeft het illegale en clubjes-circuit hier de zaak al danig verpest, en zijn er dus nauwelijks software-huizen, die legaal hun spullen in ons land te koop aanbieden. Commodore treedt hier ook nauwelijks tegen op, het lijkt wel of ze het knipogend wel leuk vinden. Maar dat is echt niet in het belang van de gebruiker, die dan met een machine zit, waarvoor geen software, geen bladen en geen support te vinden is.

U.S.

Het Geos-systeem haalt zijn kracht vooral uit het verzorgde, gestroomlijnde uiterlijk, en de snelle en handige toepassing van machinetaalroutines. Juist door dat gestroomlijnde machinetaal-gebruik is het moeilijk om zelf wat in het systeem te sleutelen. In dit artikel gaat Peter Boncz in op het zelf ontwerpen van Ikonen voor eigen programma's.

Ikonen in GEOS ontwerpen



Het woordenboek zegt: Een Ikoon is een Russische oud-christelijke schildering of mozaiek. In dit geval gaat het echter om plaatjes, in gebruiksvriendelijke computersystemen. Veelal in combinatie met een muis, is het mogelijk door de simpele handeling van het aanwijzen van het plaatje, een functie of bestand op te roepen. De muis, ikonen, verder trekmenus e.d. dienen als 'user interface', een makkelijk te gebruiken verbinding tussen mens en computer.

In het GEOS systeem worden ikonen veelvuldig gebruikt. Na het aanwijzen moet zo'n plaatje vaak worden verplaatst over het beeld. Als het verplaatst wordt is een sprite natuurlijk de handigste oplossing. Ikonen hebben dus het formaat van een sprite. Net als een sprite bestaat een ikoon dus uit 21*24 puntjes, in 21*3 = 63 bytes. Allereerst gaan we dus een sprite ontwerpen die later als ikoon voor een plaatje zal gaan dienen.

Het ontwerpen van een Ikoon

Teken een veld met 24 vakjes in de breedte en 21 vakjes in de lengte. Breng hier de gewenste tekening op aan, door de vakjes in te kleuren. Als de tekening af is, trekt u steeds na 8 hokjes, een dikke vertikale streep. De tekening is nu in drie kolommen van 8 hokjes onderverdeeld. Ieder regel van een kolom is een byte, in totaal heeft u er dus 3*21 is 63. De waarde van iedere byte berekent u zo: ieder hokje van een byte heeft een aparte waarde, deze is afhankelijk van de plaats. Helemaal links een hokje vol betekent 2 tot de macht 7 (128), een naar rechts twee tot de macht 6 (64) etc. tot twee tot de macht 0 (1). Als een hokje leeg is, betekent dat een waarde 0. Door de waardes van iedere hokje op te tellen komt u tot een eindandwoord voor ieder byte. Vervolgens neemt u de volgende byte (rechts). Als u 3 bytes gehad hebt gaat u naar onderen, totdat u alle data gevonden hebt.

De nu ontworpen sprite moet op de disk waar uw programma op staat weggeschreven worden. Om het hoe en waarom hiervan een beetje duidelijk te maken zal ik eerst wat vertellen over het disk-systeem waar GEOS mee werkt.

Diskmonitor

wanneer men de werkwijze van het GEOS systeem een beetje wil begrijpen, moet men goed in het achterhoofd houden dat het systeem in zijn totaliteit veel te groot is om in het geheugen op te slaan. Er worden dus steeds stukjes van disk ingeladen, zodat het niet mogelijk is bijvoorbeeld GeoPaint in te laden, te wijzigen, en vervolgens weer te saven.

Alleen via een diskmonitor kan men dan direkt met het systeem communiceren

Er moet dus rechtstreeks op de disk gewijzigd worden. Daarvoor is een zogenaamde diskmonitor nodig. Dit is een programma, dat gegevens direct van disk leest, de gelegenheid biedt ze door te kijken, eventueel te wijzigen en later weer weg te schrijven. In de bijgaande listing vindt u er een,

```
disk-monitor v1.0
>r 12 01
>m 00 a8
>:00   12 09 c3 01 10 47 45 4f   ..C..geo
>:08   50 41 49 4e 54 a0 a0 a0   paint
>:10   a0 a0 a0 a0 a0 01 08 01        ...
>:18   06 56 04 07 0c 00 77 00   .V....W.
>:20   00 00 c3 06 0b 47 45 4f   ..C..geo
>:28   57 52 49 54 45 a0 a0 a0   write
>:30   a0 a0 a0 a0 a0 06 03 01        ...
>:38   06 56 04 07 0c 00 58 00   .V....X.
>:40   00 00 c3 0a 13 44 45 53   ..C..des
>:48   4b 20 54 4f 50 a0 a0 a0   k top
>:50   a0 a0 a0 a0 a0 0a 0b 01        ...
>:58   04 56 04 07 0c 00 48 00   .V....h.
>:60   00 00 83 0e 05 48 61 72   .....hAR
>:68   6d 6f 6e a0 a0 a0 a0 a0   MON
>:70   a0 a0 a0 a0 a0 0e 12 01        ...
>:78   08 56 04 07 0c 00 11 00   .V......
>:80   00 00 83 0f 06 54 65 6c   .....tEL
>:88   65 67 72 61 70 68 a0 a0   EGRAPH
>:90   a0 a0 a0 a0 a0 0f 13 01        ...
>:98   08 56 04 07 0c 00 0b 00   .V......
>:a0   00 00 83 1a 07 43 61 6c   .....cAL
>█
```

```
>r 01 08
>m 00 b0
>:00  00 ff 03 15 bf ff ff ff
>:08  80 00 01 81 ff 81 87 00
>:10  e1 8c 00 39 89 00 dd 9a
>:18  c4 cf 93 0c 0f a0 1c 67
>:20  a2 1c 77 a1 38 37 a2 61
>:28  87 90 c3 c7 91 81 cf 8b
>:30  38 8f 86 7c 3d 8d f3 f9
>:38  98 e1 e1 b0 00 01 80 00
>:40  01 ff ff ff 83 06 01 00
>:48  04 ff 03 00 04 67 65 6f
>:50  50 61 69 6e 74 20 20 20
>:58  20 56 31 2e 31 00 00 00
>:60  00 4a 69 6d 20 44 65 46
>:68  72 69 73 63 6f 00 00 00
>:70  00 00 00 00 00 00 00 00
>:78  00 00 00 00 00 00 00 00
>:80  00 00 00 00 00 00 00 00
>:88  00 27 8d 84 01 01 00 00
>:90  a9 49 85 0f a9 6f 85 0e
>:98  a9 05 85 10 a9 08 85 11
>:a0  67 65 6f 50 61 69 6e 74
>:a8  20 69 73 20 61 20 66 75
>
```



maar als u zelf over een andere beschikt, kan die natuurlijk ook voldoen. De Commodore 1541 diskdrive formatteert een diskette in 35 tracks, die onderverdeeld zijn in sectoren. Track 1 tm 17 hebben 21 sectoren (genummerd van 0 to 20), track 18 tm 24 hebben 19 sectoren, track 25 tm 30 hebben er 18 en 30 tm 35 hebben er 17. Iedere sector, ook wel blok genoemd bestaat uit 256 onderdeeltjes (bytes). Een byte bestaat uit een getal van 0 tot 255. Deze getallen zijn de gegevens waaruit een programma bestaat.

Track 18

In de directory staat, welke programma's zich op disk bevinden en op welke plaats van de disk zij staan. Tevens bevinden zich daar gegevens als de grootte van de file en het soort bestand (seq, prg, usr of rel). De directory staat altijd in die haast magische track 18. In sector 0 is de disknaam en -ID vastgelegd. In sector 1 begint de echte directory. Vaak is die echter groter als 256 tekens, zodat hij in een andere sector voortgezet moet worden. In welke track en sector dat is, staat in de eerste twee bytes van een sector. Dit is erg belangrijk voor het werken met de diskmonitor.

Onze diskmonitor kent 3 commando's: **R tr se** (Lees Tracknummer Sectornummer), **W tr se** (Schrijf Tracknummer Sectornummer), en **M bb eb** (Laat geheugen zien vanaf byte Tot byte).

De werkwijze is als volgt: laad een sector, laat dan met M de inhoud van het geheugen zien. U ziet dan de inhoud van de sector voorbijscrollen. Helemaal links staat het nummer van de getoonde byte (dit varieert van 0 tot 255 in stappen van 8). Vervolgens komt de inhoud in getallen, en daarnaast wordt, indien er tekst aanwezig is, deze in karakterstrings zichtbaar gemaakt.

Denkt u er wel steeds aan dat de getallen die diskmonitor laat zien altijd hexadecimaal (16-tallig) zijn, dus van 00 (0*16 + 0) tot FF (15*16 + 15). Dit in tegenstelling tot het gebruikelijke 10-tallige stelsel, daar betekent 41: 4*10 + 1, in hexadecimaal zou het dus betekenen 4*16 + 1, een heel verschil dus!!! Voor een 16 tallig stelsel heb je natuurlijk ook 16 cijfers nodig. Aangezien die er niet zijn heeft men naar letters gegrepen. De letters A tm F stellen daarom de waardes 10 tm 15 voor. Omrekenen van hexadecimaal naar tientallig (normaal) betekent dus: de waarde van het linkse cijfer annex letter vermenigvuldigen met 16 en er de waarde van het tweede cijfer bij optellen.

Werkwijze

Terug naar de monitor nu. U had bijvoorbeeld ingetikt R 12 01 (lees track 18 sector 1) en vervolgens M 00 FF (de hele sector op beeld) Om de in het geheugen staande sector te veranderen moet u de waardes van de bytes (en dus niet de karakters) veranderen. Als de eventuele veranderingen in een sector zijn aangebracht, is het raadzaam deze weer op disk te zetten, anders gaan uw wijzigingen verloren. Dit doet men met het al besproken commando W (write), in ons geval dus W 12 01.

Wanneer we nu de directory van een GEOS disk bekijken (track 18 sector 1), zien we in een oogopslag de namen van de files in de karakterstrings staan. De getallen voor de letters stellen natuurlijk dezelfde gegevens voor; als die waardes veranderd worden, veranderen ook de letters. Het is nu belangrijk te letten op de eerste twee data voordat de naam begint. De plaats is altijd de 4e en 5e data van links op een regel. Deze twee geven respectievelijk de track en sector aan waar de file op disk staat. Een ander belangrijk adres bevindt zich na de hoeveelheid lege bytes, die achter de naam staat (gevuld met de waarde A0). Na deze bytes (de 16e na het begin van de programmanaam) staat weer een adres van een track en sector (altijd op de 6e en 7e plaats van links). Dit noem ik zelf de subdirectory, hierin staan gegevens over de GEOS file. Bij gewone BASIC programma's zijn deze waarden niet gevuld, en de file heeft ook geen subdirectory.

Subdirectory

We lezen nu een subdirectory sector in en nemen de eerder ontworpen spritedata ter hand. Deze plaatsen we (in HEXADECIMALE getallen) over de oorspronkelijke waardes van de file, te beginnen bij de 6e byte. Hier staan namelijk altijd de normale Ikoon data. Vergeet niet op return te drukken na iedere regel en vergeet vooral niet de sector na afloop weer weg te schrijven, anders zouden uw veranderingen geen effect hebben.

In het geval dat een file nog geen subdirectory had, kunt u er natuurlijk geen wijzigen, maar moet er een nieuwe gemaakt worden. Het is dan het handigste een subdirectory van een of andere GEOS file te nemen (=in te lezen), de sprite-data te wijzigen en de sector weer op een vrije plaats op uw eigen disk weg te schrijven. Als u geen lege plekken op uw disk weet, kunt u er altijd wel een in track 18 (HEX 12) vinden, want de directory neemt nooit veel ruimte in beslag . Vergeet dan ook niet achter de programmanaam in de gewone directory het adres van de subdirectory te zetten.

Als u dit allemaal gedaan hebt, zult u, in GEOS het resultaat van uw noeste arbeid zien. Veel succes ermee!

Peter Boncz

LISTING PROGRAMMA :

```
  5 FORA=0TO735:READB:POKE49152+A,
    B:T=T+B:NEXT
  6 IFT<>90444THENPRINT"FOUT IN DA
    TA":END
  7 PRINT"[CLR-HOME][3XNEER]SYS491
    52[HOME]":END
 10 DATA 162,0,189,133,194,32,210,
    255,232,224,18,208,245,162,13,
    169,62,32
 11 DATA 235,192,169,0,141,1,2,32,
    51,193,201,62,240,249,201,32,2
    40,245
 12 DATA 162,5,221,106,192,208,12,
    142,0,2,189,112,192,72,189,118
    ,192,72
 13 DATA 96,202,16,236,76,13,192,1
    33,151,32,98,192,185,224,194,3
    2,220,192
 14 DATA 200,208,3,238,1,2,198,151
    ,208,237,96,32,254,192,144,3,1
    53,224
 15 DATA 194,200,198,151,96,32,98,
    192,169,32,44,169,13,76,210,25
    5,58,87
 16 DATA 82,77,64,88,192,193,193,1
    92,193,227,192,144,144,123,62,
    122,160,0
 17 DATA 140,3,2,136,140,4,2,32,20
    7,255,201,13,240,23,32,254,192
    ,144
 18 DATA 18,141,3,2,32,207,255,201
    ,13,240,8,32,254,192,144,3,141
    ,4
 19 DATA 2,172,3,2,32,198,194,32,2
    14,194,152,32,220,192,32,98,19
    2,169
 20 DATA 8,32,61,192,32,151,194,76
    ,166,192,76,13,192,32,254,192,
    144,248
 21 DATA 168,169,8,133,151,32,51,1
    93,32,51,193,32,83,192,208,248
    ,32,151
 22 DATA 194,76,13,192,72,74,74,74
    ,74,32,244,192,170,104,41,15,3
    2,244
 23 DATA 192,72,138,32,210,255,104
    ,76,210,255,24,105,246,144,2,1
    05,6,105
 24 DATA 58,96,169,0,141,2,2,32,51
    ,193,201,32,208,9,32,51,193,20
    1
 25 DATA 32,208,15,24,96,32,40,193
    ,10,10,10,10,141,2,2,32,51,193
 26 DATA 32,40,193,13,2,2,56,96,20
    1,58,8,41,15,40,144,2,105,8
 27 DATA 96,32,207,255,201,13,208,
    248,104,104,76,13,192,32,207,2
    55,201,13
 28 DATA 208,39,169,0,133,144,32,1
    01,192,169,8,133,186,32,180.25
    5,169,111
 29 DATA 133,185,32,150,255,32,165
    ,255,36,144,112,5,32,210,255,2
    08,244,32
 30 DATA 171,255,76,13,192,201,36,
    240,29,72,169,8,133,186,32,177
    ,255,169
 31 DATA 111,133,185,32,147,255,10
    4,32,168,255,32,207,255,201,13
    ,208,246,32
 32 DATA 174,255,76,13,192,32,51,1
    93,32,254,192,144,245,141,39,1
    94,32,51
 33 DATA 193,32,254,192,144,234,14
    1,42,194,32,73,194,173,0,2,201
    ,1,240
 34 DATA 30,169,49,32,237,193,162,
    13,32,198,255,162,0,32,207,255
    ,157,224
 35 DATA 194,232,208,247,32,204,25
    5,32,110,194,76,13,192,32,44,1
    94,162,13
 36 DATA 32,201,255,162,0,189,224,
    194,32,210,255,232,208,247,32,
    204,255,169
 37 DATA 50,32,237,193,76,201,193,
    141,32,194,162,15,173,39,194,3
    2,120,194
 38 DATA 142,39,194,141,40,194,173
    ,42,194,32,120,194,142,42,194,
    141,43,194
 39 DATA 162,15,32,201,255,162,0,1
    89,31,194,32,210,255,232,224,1
    3,208,245
 40 DATA 76,204,255,85,49,58,49,51
    ,32,48,32,0,0,32,0,0,162,15
 41 DATA 32,201,255,162,0,189,65,1
    94,32,210,255,232,224,8,208,24
    5,76,204
 42 DATA 255,66,45,80,32,49,51,32,
    48,169,15,168,162,8,32,186,255
    ,169
 43 DATA 0,32,189,255,32,192,255,1
    69,13,168,162,8,32,186,255,169
    ,1,162
 44 DATA 109,160,194,32,189,255,76
    ,192,255,35,169,13,32,195,255,
    169,15,76
 45 DATA 195,255,162,48,56,233,10,
    144,3,232,176,249,105,58,96,13
    ,68,73
 46 DATA 83,75,45,77,79,78,73,84,7
    9,82,32,86,49,46,48,152,56,233
 47 DATA 8,168,32,98,192,169,18,32
    ,210,255,162,8,185,224,194,41,
    127,201
 48 DATA 32,176,4,169,46,208,3,185
    ,224,194,32,210,255,169,0,133,
    212,200
 49 DATA 202,208,229,169,146,76,21
    0,255,173,1,2,208,6,204,4,2,17
    6,1
 50 DATA 96,104,104,76,13,192,32,1
    01,192,169,58,162,62,76,235,19
    2
```

*** *EINDE LISTING* ***

```
REGEL 5    85
REGEL 6    117
REGEL 7    116
REGEL 10   218
REGEL 11   47
REGEL 12   119
REGEL 13   234
REGEL 14   69
REGEL 15   162
REGEL 16   243
REGEL 17   191
REGEL 18   100
REGEL 19   30
REGEL 20   143
REGEL 21   132
REGEL 22   33
REGEL 23   154
REGEL 24   67
REGEL 25   245
REGEL 26   119
REGEL 27   207
REGEL 28   227
REGEL 29   208
REGEL 30   197
REGEL 31   53
REGEL 32   183
REGEL 33   56
REGEL 34   181
REGEL 35   220
REGEL 36   61
REGEL 37   226
REGEL 38   19
REGEL 39   203
REGEL 40   130
REGEL 41   170
REGEL 42   7
REGEL 43   93
REGEL 44   5
REGEL 45   97
REGEL 46   93
REGEL 47   140
REGEL 48   115
REGEL 49   170
REGEL 50   97

READY.
```

Rob Goudriaan bespreekt weer een aantal klassieke spellen. Deze toppers uit het verleden zijn soms nog te koop, naar anderen zult u echt moeten zoeken.

Oud van GOUDriaan

Cohen's Towers

Onze held in dit spel is Allen, het neefje van de baas. Hij krijgt de opdracht om pakjes te verzamelen in een 17 etages tellend gebouw.

Per etage staan er één of twee pakjes. Je kan maar één pakje per keer dragen, zodra je een pakje opgepakt hebt moet je die naar één van de drie aanwezige postbussen brengen. Pas op dat je niet te lang in de lift blijft, want de baas houdt van opschieten, gaat het niet hard genoeg dan krijg je een bloempot op je kop, wat niet zo leuk is, want je duikelt meteen naar de onderste etage. Blijf ook niet te lang op een etage lopen want dan komt er een kat die je er wel even vanaf gooit, met als resultaat dat je weer helemaal onderaan moet beginnen. Je kunt wel over de kat heenspringen maar dan moet je daarna wel heel snel doorlopen. Ook zijn er nog ronddolende dames, wat daar de bedoeling van is moet U zelf maar ontdekken. Punten kun je ook verzamelen door vanuit de lift snel op de tussenbalken te springen. Per keer verdien je 50 punten, maar je moet wel schuilen voor de vallende bloempotten. Het pakken van een sleutel levert de meeste punten op. Cohens Towers is een spel waarbij men snel moet kunnen reageren, en dat je vaak uit de kast zult halen.

Hunchback

Lopend, springend en zwaaiend gaat u over een kasteelmuur om Jane uit de handen van een slechte kasteelheer te redden. Maar voor U zover bent moet er wel het één en ander gedaan worden.

De eerste hindernis die u moet overwinnen zijn de grote rollende ballen die over de muur rollen en al springend bereikt u de eerste bel. Deze bellen zijn erg belangrijk want heeft u er vijf bij elkaar dan levert dat een aardige bonus op. Als je de rollende ballen voorbij bent sta je voor een grote gracht, gelukkig hangt er een touw, even denken dat u Tarzan bent, in het touw klimmen en naar de overkant zwaaien. Vergeet niet er op tijd af te springen anders blijf je heen en weer zwaaien. Daarna volgt een muur met kantelen, die niet echt moeilijk is, zeker niet als je weet wat er daarna nog komt.

De moeilijkheden beginnen pas goed op de muur, waar wachters met scherpe spiezen niets liever doen dan jou aan hun spies te prikken. Of dat al niet erg genoeg is beginnen ze bij het volgende niveau ook nog met kogels te gooien, natuurlijk net op het moment dat jij over een man heenspringt. Wat verderop staan er u nog speren te wachten die afwisselend hoog en laag, van voren en van achteren komen.

Al met al een hele klus om goed over al die hindernissen te komen. We verklappen natuurlijk niet alles, er moet tenslotte ook voor u nog wat te ontdekken overblijven. Het zal U zeker vele uren kosten om de jonkvrouw te redden.

Archon

Archon is een strategiespel van Electronic Arts. Archon is een soort schaakspel maar met extra dimensies, het spel wordt gespeeld door twee partijen, namelijk licht en duisternis, die elk 18 stukken hebben.

De namen komen voornamelijk uit de mythologie. Aan het hoofd van de twee partijen staat een tovenaar (of tovenares), die over geheimzinnige krachten beschikt zoals bijvoorbeeld: teleporteren, bevriezen, het aanroepen van natuurkrachten enz.

Het spel wordt gespeeld op een bord van 9x9 vakken en aan beide zijden staan de 18 stukken opgesteld. Het doel van het spel is het veroveren van de vijf krachtvelden of het vernietigen van de tegenstander. Deze krachtvelden zijn herkenbaar aan een witte stip. Het belangrijkste verschil met schaken is dat een stuk dat geslagen is niet direkt van het bord verdwijnt maar op een apart scherm moet gaan uitvechten wie de sterkste is, de sterkste speler krijgt het veld waar voor gevochten is. De witte stukken zijn het sterkst op de lichte velden, de zwarte op de donkere. Gekleurde velden veranderen na iedere ronde van kleur. Het spel wordt met de joystick gespeeld, tegen de computer of tegen een andere tegenstander.

Archon is geen spel dat je even leert spelen, er is veel ervaring voor nodig om de stukken snel te herkennen en op de beste, sterkste plaats te zetten. Het tweede scherm is buiten de stippen geheel zwart. Deze stippen dienen als schild tijdens het gevecht maar pas op, zij verwisselen nog wel eens van plaats. Aan de zijkanten geven twee balken de kracht aan van de stukken, bij elke treffer worden ze korter. Na het gebruik van een wapen duurt het even tot de energie weer terug komt en de enige remedie is dan achter de stippen weg te kruipen.

Archon is een moeilijk maar erg boeiend spel waarin men steeds weer wat nieuws in ondekt. Vooral met de computer als tegenstander is het een spel dat je vaak verliest, maar dat maakt het juist zo aantrekkelijk.

Impossible Mission

Letterlijk vertaald heet dit spel: onmogelijke opdracht. Zoals dus al uit de naam is af te leiden is dit spel (bijna) niet te voltooien, maar voor de volhouders zijn er toch wel mogelijkheden, ondanks de ongeveer 93 robots die er van alles aan doen dit te voorkomen.

Deze robots krijgen aan het begin van het spel voor iedereen duidelijk hoorbaar de opdracht van hun professor de indringer (u dus) te vernietigen. De opzet van het spel is het vinden van de controlekamer van de professor. Dit is alleen mogelijk door het vinden van een codewoord in een grotstelsel met ruim 32 kamers. In deze kamers staan bedden, kasten, bureau's, prullebakken enz. De bedoe-

ling is dat u overal in, onder, achter kijkt of er een van de 36 puzzelstukjes is verstopt. Deze stukjes worden automatisch overgebracht naar een computerruimte.
Om een robot te misleiden zijn er diverse mogelijkheden, snelle en onverwachte bewegingen maken, of over de robot heen springen. Ook zijn er twee kamers met een schakelkast, deze speelt een melodie die men na moet spelen, als dat lukt verdien je een *password* dat ook als hulp tegen de robots gebruik kan worden. Hiermee kun je namelijk de robots voor eventjes laten inslapen of een lift in de startpositie terugbrengen. Een centrale lift verzorgt de verbinding tussen de diverse kamers. Met de joystick kun je je mannetje een salto laten maken waardoor het over gapende afgronden springt. Zet het geluid, zeker als u gehorig woont, niet te hard, anders heb je binnen de korste keren de politie op de stoep, gewaarschuwd door bezorgde buren.
In de centrale computerkamer waar je komt door buiten de kamers op de vuurknop te drukken, kun je de verzamelde stukjes vergelijken, draaien, van kleur veranderen. Het is hier ook mogelijk om te bellen en advies aan de computer te vragen. Vier stukjes die volledig over elkaar heenvallen leveren een letter op, het code woord bestaat uit negen letters. Maar let wel op de tijd want die loopt door. Elk leven dat men verliest kost 10 minuten van de speeltijd.
Als u niet op tijd de oplossing heeft gevonden wordt u op een honende manier uitgelachen door onze professor. Elke keer dat men het spel opnieuw start is de kamerindeling en de volgorde van de puzzelstukjes weer anders.
Het spel heeft alle mogelijkheden van de Commodore volledig benut met mooie grafische plaatjes en goede geluidseffecten.

Frogger.

Waarom het ene spelletje wel en het andere spelletje niet aanslaat is voor mij nog steeds een raadsel, ligt het aan de muziek, misschien wel aan de grafische opbouw of de gebruikte sprites.

Smaken verschillen natuurlijk maar ineens is er dan weer een spel dat iedereen eigenlijk wel leuk vindt. Zo'n spel was Frogger, een spelletje uit het prille begin. Eigenlijk een vreemd spel, een kikker die niet kan zwemmen en achtergrondmuziek die vrij eentonig is. Onze kikker zit aan de rand van een snelweg waar allerlei vreemdsoortig verkeer heen en weer rijdt op een vierbaansweg. Elke opening tussen de voertuigen moet je benutten om zo snel mogelijk naar de overkant te springen. Een klein tikje tegen de joystick is al voldoende om de eerste bewegingen te maken. Ter begeleiding of aanmoediging is er het hoempa-muziekje. Frogger moet soms de raarste sprongen maken want erg veel ruimte is er niet om over te steken. Na de weg te zijn overgestoken sta je aan de rand van een rivier. Normaal gesproken is voor kikkers de meest logische weg: door de rivier. Voor Frogger niet, die heeft als motto: doe wat moeilijk kan niet makkelijk. Dus de weg gaat via boomstammen en schildpadden die af en toe blijken onder te duiken, dus goed opletten want daar kan onze kikker niet tegen. De bedoeling is dat men via deze boomstammen en schildpadden veilig aan de overkant komt en in een poortje duikt dat nog vrij is. In totaal moeten er per ronde 5 kikkers naar de overkant gebracht worden. In de tweede ronde lijkt het spitsuur op de snelweg. De procedure is hetzelfde als in de eerste, alleen gaat alles veel sneller. De boomstammen zijn minder lang en er moet extra opgepast worden want nu zijn er ook als boomstam vermomde krokodillen. Je kunt wel op zijn rug springen maar zelfs onze niet-zwemmende kikker snapt dat je niet bij zijn bek moet komen.
Als U al deze gevaren getrotseerd heeft belanden we in ronde drie, waar alles nog sneller gaat. Als extra handicap wordt nu ook de tot nu toe veilige middenberm onveilig gemaakt door een slangetje van het gevaarlijkste soort. Aanraking is dodelijk. Op de boomstammen huppelen nu levensgevaarlijke otters rond. Het is dus zaak om zeer snel naar de overkant te gaan, want er moeten nog steeds 5 kikkers overgebracht worden. De schildpadden duiken niet meer onder water, hierop kun je dus even uitrusten. Van nu af wordt het spel niet moeilijker meer, het enige dat nu nog telt is het uithoudingsvermogen zodat je betrekkelijk rustig kan werken aan een topscore.

Q Bert.

Aanhangers van dit spel zijn er denk ik net zoveel als tegenstanders. Waarschijnlijk is dit het spel dat het meeste is geïmiteerd, maar dit kan de trouwe aanhangers geen kwaad doen.

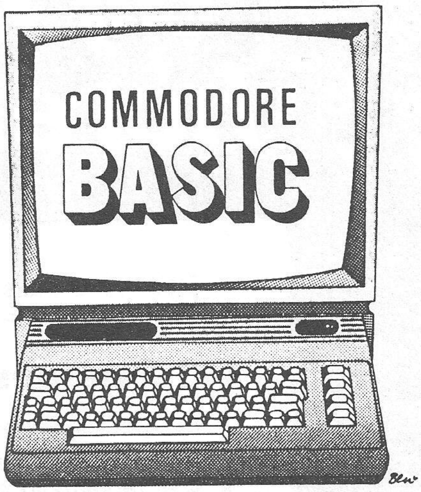
Zelf spring ik altijd alle kanten op behalve de goede. Spring ik eindelijk eens op het goede blokje dan springt één van die engerds boven op me. Na geduldig gekeken te hebben bij betere spelers ben ik tot de conclusie gekomen dat ik het Q-Bert fort maar beter door anderen kan laten oplossen. Ik begin dan te genieten als ik kan zeggen: alweer ernaast. Waarna ik al menigmaal vriendelijk doch dringend ben verzocht de kamer te verlaten.
De bedoeling van het spel is dat je in een driedimensionale driehoek van blokje naar blokje springt. Zodra je op zo'n blokje komt verkleurt dit. Om het geheel nog moeilijker te maken springen er allerlei rare wezens rond, waaronder rode en paarse ballen. Ook krijg je te maken met dodelijke, omhoog kruipende wezens. Aan de zijkant zijn twee ronddraaiende schijven waarop je kunt springen. Hierdoor wordt je weer naar boven gebracht. Dit is een goede manier om achtervolgers van je af te schudden, want deze sprong loopt voor hun altijd fataal af. Het is maar goed dat dit een spel uit de eerste periode is, toen de 64 nog niet zo goed kon spreken, want als ik zie hoe Q Bert kan schelden als hij wordt gevangen! ●

Voor iedereen die wat meer wil doen met z'n Commodore dan het runnen van de standaard spelletjes en commerciële programma's, of het nu met de VIC, C-64, C-128 of de PC 10/20 met GWBASIC gebeurt, is deze serie artikelen over de meeste Basic-begrippen geschreven. Stap voor stap, of -beter wellicht- 'Byte voor Byte', worden hier de meest gebruikte Basic-opdrachten behandeld en toegepast op een manier die vooral bestemd is voor de nieuwkomers op dit gebied. Maar ook de wat gevorderde Basic-programmeur kan z'n hart en kennis ophalen bij deze basis cursus van Jan Bodzinga.

Basis Basic

Deel 13

Een Database (II)



In aflevering 12 van deze cursus zijn we begonnen met een vrij lang en ingewikkeld Basicprogramma, wat ons uiteindelijk een bijna professioneel stuk software zal opleveren, waarmee we een krachtige 'database' kunnen beheren. Vooral de principes van dit gegevensbestand komen aan de orde. Aan het einde van dit artikel zijn we weer een heleboel wijzer over de manier waarop een database in Basic kan worden aangemaakt en onderhouden. En tussen de tekst door bouwen we verder aan ons eigen programma, waarin allerhande zaken uit eerdere lessen nog weer eens boven de tafel worden gehaald.

Herhaling

De vorige aflevering hebben we afgesloten met een gedeelte van de listing van ons database-programma. We hebben de nodige data-regels te verwerken gekregen, terwijl ook het eerste gedeelte van het **hoofdmenu** al is gepubliceerd. Om de hele zaak nog even in ons eigen miljoenen-bits-tellende geheugen terug te halen zullen we beginnen met de nog ontbrekende listingregels van het eerste programma-deel:

```
10   REM Menu gestuurde
     DATABASE
20   REM voor adressenbestand
30   REM Commodore-Info 1987
40   REM 870228/ v.1.02
100  REM initialisatie
110  REM
120  TT=0 : REM totaal aantal
     records
130  DIM A$(300,5) : REM aantal
     records max.
140  KZ=0 : REM keuze
150  I = 0 : REM temp.var
160  DIM KZ$(10) : REM keuze-
     strings
170  DIM ST((LOG(300)/LOG(2)+4)
     , 1) :REM TEMP-ARRAYS
180  DIM VN$(7) : REM veldnamen
200  REM init
210  GOSUB 900 : REM keuze lezen
299  GOTO 1000 : REM begin
     programma
```

De laatste drie regels zorgen voor de initialisatie van de nodige programma-variabelen, die niet direct tot de standaardgegevens van het database-programma behoren. Als eerste wordt in regel 210 de subroutine van regel 900 en verder uitgevoerd. In die routine werden de data-regels ingelezen, die de vaste veldnamen bevatten:

```
899  REM inlezen keuze-
     mogelijkheden
900  I = 0 : TK=0
910  READ KZ$(I)
920  IF KZ$(I) = "eind" THEN
     TK = I - 1 : GOTO 950
930  I = I + 1
940  GOTO 910
950  FOR I = 0 TO 5
960  READ VN$(I)
970  NEXT I
```

De data die we lezen in de variabele-arrays **KZ$()** en **VN$()** kunnen we gemakkelijk naar eigen ideeën aanpassen, zoals de vorige keer uitvoerig aan de orde is geweest. Niet alleen vinden we hier de namen voor de velden (420 - 425) maar ook de keuze-omschrijvingen die in het menu aan de orde komen:

```
399  REM data keuzemogelijkheden
400  DATA "KEUZEMENU ADRES-
     SENBESTAND"
410  DATA "Records toevoegen"
411  DATA "Records wijzigen"
412  DATA "Records verwijderen"
413  DATA "Records opzoeken"
414  DATA "Bestand sorteren"
415  DATA "Bestand printen"
416  DATA "Bestand inlezen"
417  DATA "Bestand wegschrijven"
```

```
418 DATA "Stoppen"
419 DATA "eind"
420 DATA "Naam          "
421 DATA "Voorletters   "
422 DATA "Straat en nummer  "
423 DATA "Postkode      "
424 DATA "Woonplaats    "
425 DATA "Telefoon      "
```

Op regel 300 staat een korte routine die op veel plaatsen in het programma wordt aangeroepen. De regels zorgen er voor, dat er vanuit de tijdelijke array **HA$()** de verschillende **velden** worden weggezet in de kaartenbak-array **A$(X,Y)**. Deze array bevat alle adressen in twee dimensies:

```
300 REM var. wegzetten in array
310 FOR II = 0 TO 5
320 A$(AA,II)=HA$(II)
330 NEXT
340 RETURN
```

In deze routine worden de **velden**, ofwel de tweede dimensie van array **A$(X,Y)** gevuld met de gegevens van array **HA$()**. We doen dit met een FOR..TO..NEXT loop, waarbij de teller **AA** de juiste **index** van het adresnummer in de array aangeeft, terwijl de verschillende velden door de lusteller **II** worden weergegeven. De (tijdelijke) array **HA$()** bevat slechts zes indices, dus voor ieder veld één index.
Verderop in de listing zullen we een beter inzicht krijgen in het hoe en waarom van deze routine.

De TAB ( ) functie

Hoewel we gaandeweg al heel wat weten over Basic, zijn er toch nog altijd een stel leuke functies die we nog niet echt behandeld hebben. Twee ervan zitten verstopt in dit programma. Dat zijn de functies **TAB( )** en **LEN( )**. Niet erg moeilijk te begrijpen, maar wel leuk en functioneel om te gebruiken.
TAB( ) is een zogenaamde stringfunctie, omdat de waarde die na het uitvoeren van de TAB-functie wordt teruggestuurd naar de computer van het formaat '**string**' is. In feite wordt een aantal spaties teruggegeven, maar ook spaties horen tot de stringcategorie. De TAB( ) functie beweegt de cursor (of de printkop) een **relatief** aantal spaties verder. Het aantal wordt aangegeven door een numerieke waarde, die bij de TAB( ) tussen de haken wordt gezet. Probeer de volgende regels maar eens uit:

```
10 PRINT TAB(5) ;"Info"
20 PRINT TAB(8) ;"Commodore"
30 PRINT TAB(3);"Info"; REM
   NB PUNTKOMMA
40 PRINT TAB(9);"Commodore"
```

Of deze :

```
10 FOR I = 1 TO 40 step 3
20 PRINT TAB (I) "INFO"
30 NEXT I
```

Met een beetje fantasie zijn zelfs met deze eenvoudige printfunctie leuke schermcapriolen uit te halen, terwijl het zeker aardige resultaten oplevert, als je daarbij ook nog verschillende kleuren gaat gebruiken.
Let er wel op, dat **TAB( )** altijd begint te rekenen vanaf de meest linkse kolom van de schermregel, maar vervolgens relatief werkt. Dat betekent dat op één regel alle **TAB( )**'s en tussenliggende karakters bij elkaar worden opgeteld, zelfs zover, dat het als het nodig is, op de volgende regel wordt verder geTABt.
De numerieke waarde die aan TAB( ) kan worden meegegeven moet liggen tussen 0 en 255.
Helaas is het niet mogelijk om de TAB-functie te gebruiken in combinatie met de opdrachten om gegevens naar bestanden te PRINTen. De opdracht **PRINT#2, TAB (5)** haalt dus weinig uit. Ook bij het **printen** op papier kan het voorkomen, dat **TAB** niet goed samenwerkt met de printer. Dan moet er gebruik worden gemaakt van de **SPC( )** functie.

LEN( )-functie

De LEN( )-functie gebruiken we om de lengte te bepalen van een **string**. In tegenstelling tot wat je misschien verwacht is **LEN( )** nu juist een functie van het **integer-type**. Dit wil zeggen, dat er een integer getal door deze functie wordt teruggestuurd naar de computer. Vergelijk dat eens met **TAB( )**, waar juist een string retour kwam.
De string waarvan we de lengte willen weten, inclusief spaties, moeten we als tekst of variabele tussen de haken van de **LEN( )** opdracht plaatsen.

LEN("COMMODORE ") en
LEN(QQ$)
zijn korrekte opdrachten, terwijl

LEN(234) en
LEN ('DDDD ')
LEN "ERSDFS"
natuurlijk een 'TYPE MISMATCH of 'SYNTAX ERROR' zullen geven tijdens het runnen. Ook de niet te 'printen' (controle)karakters worden bij LEN( ) netjes meegeteld.

In het volgende stuk van onze database worden zowel de **TAB( )** als de **LEN( )** functie gecombineerd gebruikt:

```
500REM printen kop
510 PRINT CHR$(147) : REM CLS
520 PRINT "************************
    ****************"
530 PRINT: PRINT TAB(20 - LEN
    (KZ$ (KZ)) /2);KZ$(KZ):PRINT
540 PRINT "Aantal records : ";TT
550 PRINT
    "**********************************
    *******"
560 PRINT
570 RETURN
```

Een simpel stukje Basic-werk zit in deze routine van regel 500 tot 570. In dit stukje database-programma wordt niet veel anders gedaan dan het (opnieuw) geheel schoonmaken van 't scherm, door het printen van **CLR-HOME**, op de ASCII-manier, met **CHR$(147)**, waarna een vaste **schermtitel** wordt geprint. Het aardige van deze routine is, dat er niet steeds dezelfde gegevens worden geprint. Als je de listing goed bekijkt zul je zelf gemakkelijk zien wat er aan de hand is. In regel 530 vinden we de truc:

```
530 PRINT: PRINT TAB(20-LEN
    (KZ$(KZ))/2);KZ$(KZ):PRINT
```

Als we alle PRINT-opdrachten weglaten, blijft er de volgende Basic-(functie)opdracht over die op het scherm moet worden gezet:

```
TAB ( 20 - LEN ( KZ$(KZ) ) /2) ;
KZ$(KZ)
```

De string **KZ$(KZ)** wordt geprint. Dit is de keuzestring uit het menu met de index van de gemaakte keuze (**KZ**). Als aardige toevoeging wordt deze string ook nog precies midden op het scherm geprint. Dit komt, omdat er met hulp van een **TAB( )-functie** het begin van de te printen string wordt berekend. We hebben dit zo opgelost, dat er, zonder veel gereken, altijd precies gecentreerd wordt geprint. Op zichzelf een aardige routine om ook in andere programma's toe te passen.
We gaan daarbij uit van de totale breedte van het scherm. We weten dat er 40 kolommen op een Commodore-scherm kunnen, zodat het midden van de schermregel op **40/2 = 20** posities van het begin komt te liggen. Vandaar het getal **20** in regel 530. Daarmee hebben we het precieze

centrum van de regel in de **TAB( )** aangegeven.
Als we echter de string **KZ$(KZ)** vanaf dat punt zouden printen, wordt hij niet centraal op de regel gezet. Daarom moet vanaf positie 20 nog een gedeelte naar voren worden begonnen. De berekening wordt daarom voortgezet met het uitrekenen van de totale lengte van string **KZ$(KZ)**. We hebben daar in Basic de **LEN( )**-functie voor. Door nu de verkregen lengte van de string doormidden te delen, krijgen we de helft van de hele string die op z'n beurt van de 20 posities wordt afgetrokken om de startkolom op de schermregel te bepalen.

Maak een keuze

In de volgende hulproutine wordt bekeken of de in het menu gemaakte keuze geldig is in het bestek van ons programma:

```
600  REM menu printen en keuze
610  GOSUB 500: REM printkop
620  FOR I= 1 TO TK
630  PRINT I;" - "; KZ$(I)
640  NEXT I
650  PRINT : PRINT "Maak een
     keuze ";
660  GOSUB 800 : REM lees toets
670  KZ= VAL(T$)
680  IF KZ < OR KZ >TK THEN 660
690  RETURN : REM terug met kz
```

De keuze waarmee deze routine wordt begonnen zit op dat moment in variabele **KZ**. Als eerste wordt dan de menu-kop op het scherm gezet (regel 610), waarna in de lus van regel 620 tot 640 alle mogelijke keuzes (**KZ$(I)**) worden geprint.
Vervolgens wordt de ingetypte toets vanuit subroutine 800 vv verwerkt via de variabele **T$** waarbij in regel 680 wordt getest of de ingetypte toets geldig is.
Als **KZ** kleiner is dan 1 of groter dan **TK**, het totaal aantal keuzemogelijkheden wordt de gebruiker opnieuw gedwongen een keuze te maken.
De subroutine voert daarom alleen de RETURN uit als **KZ** binnen de gegeven waarden ligt.

```
800  REM inlezen van een toets
810  GET T$ : REM lees toets
820  IF T$="" THEN 810
830  RETURN : REM terug met T$
```

Bovenstaande vier regels zorgen ervoor, dat er een toets wordt opgehaald uit het keyboard-buffer. De opdracht **GET T$** komt per seconde vele malen terug met een waarde in **T$**. Daarbij moeten we uiteraard testen of deze waarde werkelijk van een door de gebruiker ingetypte toets is, of helemaal niets. Commodore kent namelijk niet de handige functie **INKEY$** waarbij we de test op "" kunnen overslaan. Maar op deze manier maken we ook nog eens kennis met wat rudimentaire Basic-structuren. Bovendien is het helemaal niet nodig om altijd van die handige functies en opdrachten gebruik te maken. Als je de basis van een taal als Basic onder de knie hebt en begrijpt is het later veel gemakkelijker om ook in andere talen als **PASCAL, C, Machinetaal** of **LOGO** programma's te kunnen schrijven.
Nadat de toets is ingelezen in **T$** wordt deze routine afgesloten met RETURN, om bij de keuze-routine op z'n geldigheid te worden getest.

Utility

Een handige gebruiksroutine zit op de volgende regels in het programma verscholen. Het betreft een manier om van de gebruiker een van te voren gelimiteerde keuze los te krijgen. In dit geval betreft dit de keuze tussen **JA** en **NEE**.

```
850  REM JA/NEE input
852  OK = 0 : REM vlag
855  A$="NnJj": REM
     keuzemogelijkheden
860  GOSUB 810 : REM input toets
865  FOR II= 1 TO LEN(A$)
870  IF T$ = MID$(A$,II,1) THEN
     OK=1
880  NEXT II
890  IF OK THEN RETURN: REM
     met j/n
895  GOTO 860 : REM opnieuw
898  END
```

Hierbij wordt ook gebruik gemaakt van de voorgaande routine (regels 800 - 840) om een toets vanuit het toetsenbord in te lezen in de variabele **T$**. Bij het intoetsen van een J of N kan door de gebruiker zonder problemen ook van onderkasten ofwel kleine letters gebruik worden gemaakt. Dat vergemakkelijkt het programma, hoewel het programmeren van een dergelijke routine wat meer tijd vergt. Maar nu heb je de listing voor je neus en deze routine is zo geschreven dat hij zonder veel problemen in allerlei andere listings kan worden toegepast.

De modules.

Naast het eerste programmagedeelte, wat we nu in z'n geheel hebben beschreven, komt pas het feitelijke werk. Het start- of **menu-gedeelte** van dit programma kan eigenlijk als voorbeeld dienen voor elk willekeurig programma wat met keuze-mogelijkheden is opgezet. We noemen een dergelijk programma een **modulair** stuk software. In het groot worden na een menu de afzonderlijke delen of modules als zelfstandige programma's van tape of disk ingelezen en gerund waarbij het hoofdmenu als moederprogramma in het geheugen aanwezig blijft. In ons geval hebben we de **modules** gelinkt aan het menu als **subroutines**. Dat is beter voor ons overzicht en ook de meeste (huis)computers werken efficiënter op deze manier. Bovendien zijn de afzonderlijke bewerkingen nu ook weer niet zo groot, dat ze een eigen programma nodig hebben.
Zoals we in deel I van de databases al hebben gezien bestaat ons programma uit verschillende modules:

0000 - 0100	**REM programmatitel**
0100 - 0300	**Initialisatie**
0300 - 1000	**Diverse subroutines**
1000 - 2000	**Programma(start)menu**
2000 - 3000	**Toevoegen**
3000 - 4000	**Muteren**
4000 - 5000	**Verwijderen**
5000 - 6000	**Printen**
6000 - 7000	**Sorteren**
7000 - 8000	**Zoeken**
8000 - 9000	**Inlezen**
9000 - 9500	**Wegschrijven**
9500 - 9999	**Stoppen**

Al deze onderdelen zitten op een vaste plek in de complete listing. We hebben dit zo geschreven, om ook later op een snelle en handige manier wijzigingen en verbeteringen in het programma te kunnen aanbrengen. We hoeven maar op het programma-schema te kijken om te weten in welke regels we een bepaalde module moeten zoeken. Daardoor lopen we de minste kans om op een verkeerde plaats de wijzigingen in het programma aan te brengen, wat op z'n beurt weer een grote tijdwinst kan betekenen.

Toevoegen.

Om alvast met het programma aan de slag te kunnen, zullen we beginnen met de module om gegevens aan het bestand te kunnen toevoegen:

```
2000 REM toevoegen aan bestand
2010 GOSUB 500 : REM printen
     kop
2020 PRINT
2030 FOR II = 0 TO 5
2040 PRINT VN$(II);": ";
```

```
2050 INPUT HA$(II)
2060 NEXT II
2090 PRINT : PRINT "Korrekt in-
     gevoerd (J/N)";
2100 GOSUB 850 : REM JA/NEE
2110 IF T$="n" OR T$ = "N" THEN
     2000
2120 TT=TT+1:AA=TT : GOSUB
     300 : REM wegzetten in array
2130 PRINT:PRINT "Meer records
     invoeren (J/N)";
2140 GOSUB 850
2150 IF T$="n" OR T$ = "N" THEN
     RETURN
2160 GOTO 2000
```

Basic is toch geweldig nietwaar, als je ziet hoe met hulp van een **twee-dimensionale array** en een paar programma-regels op een structurele wijze gegevens in een bestand kunnen worden ingetypt. De routine om adressen te kunnen toevoegen is te verdelen in twee gedeeltes. Als eerste vinden we de werkelijke **invoer**, die loopt van regel 2000 tot 2060. In regel 2010 wordt eerst de nieuwe kop voor deze keuzemogelijkheid op het scherm geprint. Routine 500 en verder is daar goed voor. Daarna wordt in een FOR..TO..NEXT loop met teller **II** zowel de **veldnaam** op het scherm gezet als **invoer** via INPUT in de array **HA$( )** gevraagd van de gebruiker. Hier is te zien, waarvoor deze (hulp)array nodig is. We hoeven ons op deze manier nu nog niet bezig te houden met de tellers en indexen van de echte kaartenbak-array AA$( ). Deze komt pas straks aan de beurt, als de gebruiker vindt dat hij de juiste gegevens heeft ingetypt.
Na de invoer van één adres komen we in het programma bij regel 2090, waar we een prompt op het scherm krijgen waarbij wordt gevraagd of de gegevens juist zijn. We kunnen nu een J of N intoetsen voor **Ja of Nee**. De verwerking hiervan vindt plaats in routine 850/800, waarna op grond van de inhoud van **T$** ofwel opnieuw met invoer wordt begonnen in regel 2000 (antwoord NEE) ofwel de gegevens in **AH$( )** worden gebruikt om in routine 300 de echte array te vullen (antwoord JA). Let er op, dat er niet meer getest wordt op **T$="J"** of **T$="j"**. Dit is niet nodig, omdat we zeker weten, dat **T$** uit routine 850 terugkomt met slechts één van de twee mogelijkheden : J/N. Vandaar dat er slechts één test op **T$** hoeft te worden uitgevoerd.
Op dit moment in het programma weten we ook **zeker** dat de nieuwe gegevens moeten worden toegevoegd aan de originele array, zodat we nu veilig de tellers **TT** en **AA** met 1 kunnen verhogen. Dit gebeurt allemaal in regel 2120.
De laatste prompts uit deze routine wijzen door naar het hoofdmenu of geven de gebruiker de mogelijkheid nog meer adressen in te voeren zonder eerst weer een keuze te hoeven maken in het hoofdmenu.
Bekijk de invoer-routine nog eens en ontdek hoe vaak routine 850/810 in dit kleine stukje al is gebruikt. Hier vind je een goede illustratie van het nut van een subroutine. Niet alleen spaart het een forse hoeveelheid geheugenruimte, maar ook bij veranderingen in een dergelijke routine hoeven we deze slechts eenmalig aan te brengen, om in het hele programma voorzien te zijn van een nieuwe test op JA/NEE.

Deel 1

Voor het gemak volgt hieronder de complete listing van het eerste deel van ons programma:

```
10   REM Menu gestuurde DATABASE
20   REM voor adressenbestand
30   REM Commodore-Info 1987
40   REM 870228/ v.1.02
100  REM initialisatie
110  REM
120  TT=0 : REM totaalaantal records
130  DIM A$(300,5) : REM aantal
     records max.
140  KZ=0 : REM keuze
150  I = 0 : REM temp.var
160  DIM KZ$(10) : REM keuzestrings
170  DIM ST((LOG(300)/LOG(2)+4) , 1)
     :REM TEMP-ARRAYS
180  DIM VN$(7) : REM veldnamen
200  REM init
210  GOSUB 900 : REM keuze lezen
299  GOTO 1000 : REM begin program-
     ma
300  REM var. wegzetten in array
310  FOR II = 0 TO 5
320  A$(AA,II)=HA$(II)
330  NEXT
340  RETURN
360  A$(AA,5)=TL$
399  REM data keuzemogelijkheden
400  DATA "KEUZEMENU ADRESSEN-
     BESTAND"
410  DATA "Records toevoegen"
411  DATA "Records wijzigen"
412  DATA "Records verwijderen"
413  DATA "Records opzoeken"
414  DATA "Bestand sorteren"
415  DATA "Bestand printen"
416  DATA "Bestand inlezen"
417  DATA "Bestand wegschrijven"
418  DATA "Stoppen"
419  DATA "eind"
420  DATA "Naam              "
421  DATA "Voorletters       "
422  DATA "Straat en nummer  "
423  DATA "Postkode          "
424  DATA "Woonplaats        "
425  DATA "Telefoon          "
500  REM printen kop
510  PRINT CHR$(147) : REM CLS
520  PRINT"*************************
     ****************"
530  PRINT: PRINT TAB(20-
     LEN(KZ$(KZ))/2);KZ$(KZ):PRINT
540  PRINT "Aantal records : ";TT
550  PRINT"*************************
     ****************"
560  PRINT
570  RETURN
600  REM menu printen en keuze
610  GOSUB 500: REM printkop
620  FOR I= 1 TO TK
630  PRINT I;" - "; KZ$(I)
640  NEXT I
650  PRINT : PRINT "Maak een keuze ";
660  GOSUB 800 : REM lees toets
670  KZ= VAL(T$)
680  IF KZ OR KZ  TK THEN 660
690  RETURN : REM terug met kz
800  REM inlezen van een toets
810  T$ = INKEY$ : REM lees toets
820  IF T$="" THEN 810
830  RETURN : REM terug met t$
850  REM JA/NEE input
852  OK = 0 : REM vlag
855  A$="NnJj": REM
     keuzemogelijkheden
860  GOSUB 810 : REM input toets
865  FOR II= 1 TO LEN(A$)
870  IF T$ = MID$(A$,II,1) THEN OK=1
880  NEXT II
890  IF OK THEN RETURN: REM met
     j/n
895  GOTO 860 : REM opnieuw
898  END
899  REM inlezen keuzemogelijkheden
900  I = 0 : TK=0
910  READ KZ$(I)
920  IF KZ$(I) = "eind" THEN TK = I - 1
     : GOTO 950
930  I = I + 1
940  GOTO 910
950  FOR I = 0 TO 5
960  READ VN$(I)
970  NEXT I
1000 REM begin programma
1010 SU=0:KZ=0:GOSUB 600 : REM
     menu
1020 IF KZ = TK THEN 9500 : REM
     einde
1030 ON KZ GOSUB
     2000,3000,4000,5000,6000,
     7000,8000,9000
1040 GOTO 1010 : REM opnieuw menu
```

Tot zover de aflevering voor deze keer. We hebben opnieuw een gedeelte aan de database listing toegevoegd, en kunnen nu alvast een testrun op de computer uitvoeren, waarbij de adressen in de array komen te staan.
De rest van de modules zullen we de volgende keer gaan behandelen, want het is natuurlijk allemaal veel interessanter, als we de ingetypte adressen ook kunnen bewaren op disk of cassette.

Jan Bodzinga

Deze cursus heeft als doel snel te kunnen begrijpen wat er precies aan de hand is in de Commodore. Iedereen weet, dat het hele apparaat nooit zou werken zonder een microprocessor, maar hoe zo'n ding nu echt wordt aangestuurd en geprogrammeerd is een heel ander verhaal. Daarom deze cursus over de 65XX machinetaal. We nemen hier de meeste standaard machinetaal-begrippen en -opdrachten bij de kop, zodat het voor gebruik niet veel uitmaakt welk type Commodore computer je bezit, mits de microprocessor van de 65XX familie is.

Cursus MACHINETAAL

door Tjipke van der Land

deel 5 : HET GEHEUGEN

In het vorige nummer van Commodore Info hebben we onder de kop 'De microprocessor' het al even over het adresseren van het computergeheugen gehad. We zullen er in deze les gedetailleerd op verder borduren. Daarnaast zullen we ook bekijken op welke manier de computer met Bytes en bits kan rekenen. Maar eerst iets over RAM-geheugens en de adressering.

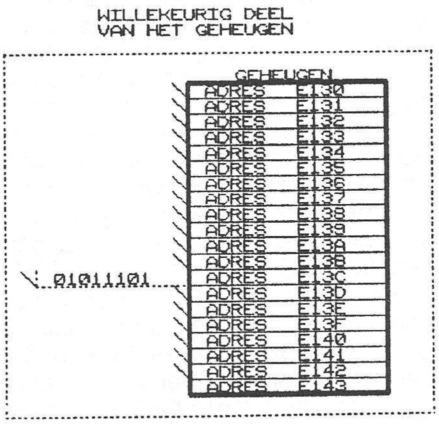
In een computer kunnen **twee soorten geheugens** voorkomen namelijk **statische** geheugens en **dynamische** geheugens. Statisch staat voor het *stabiel* zijn, het vasthouden van data; terwijl dynamisch aangeeft, dat de chip de inhoud van de geheugens nogal eens *verliest*, afhankelijk van tijdsduur en spanning. Beide soorten RAM-chips worden vaak in computers toegepast, al verdient de dynamische chip de voorkeur.

RAM-geheugen

Waarin komt het nu tot uitdrukking met welk van beide types RAM-geheugen we in onze computer te maken hebben? In de eerste plaats is de **prijs** van beide nogal verschillend, statische RAM is veel **duurder**, en is daarnaast langzamer in **toegangstijd** (access-time), maar kan dan ook weer een grotere capaciteit aan Bytes bevatten. Voornamelijk vanwege kosten en snelheid worden er echter meestal **dynamische** geheugens toegepast bij het fabriceren van computers. Pas als het echt nodig is, worden er statische chips gemonteerd. We vinden deze laatste bijvoorbeeld nog al eens in de schootcomputers, die deels op een batterijvoeding werken. Aan het gebruik van dynamische RAM zit één groot **nadeel,** de geheugen-inhoud moet **'gerefreshed'** worden, vandaar de naam dynamisch. Dat betekent dat één keer in de zoveel tijd elke geheugencel op z'n (spannings)-niveau moet worden gehouden, omdat deze cellen de eigenschap hebben van een condensator die leegloopt als je hem niet af en toe bijlaadt met een frisse dosis (nieuwe) gegevens.

Bytes

Om een RAM-geheugen te kunnen aanspreken moet er weer de ons inmiddels bekende **databus** en **adresbus** aanwezig zijn. De databus is nodig om de gegevens te transporteren en de adresbus wordt gebruikt om de geheugen-lokaties (de adressen) aan te wijzen. We werken samen met deze bussen met data-eenheden, die **Bytes** worden genoemd. **Bytes** is een begrip wat we al een aantal keren in deze serie artikelen zijn tegengekomen. Een Byte is hetzelfde als 8 bits, dus als je spreekt over een **'woord'** van 8 bits dan kun je evengoed spreken over een woord van 1 Byte. Aangezien de databus 8 bits breed is en de 65XX processor maximaal **64 KByte** kan adresseren,



Ladenkast RAM-geheugen

kun je stellen dat per adres 8 bits gelezen of geschreven kunnen worden. Na een simpele rekensom kun je zien dat deze (Commodore)-computer een totaal-geheugen heeft van **64K x 8bits = 512 kilobits** of 64K x 1Byte = 64 KByte. Nu weet je ook

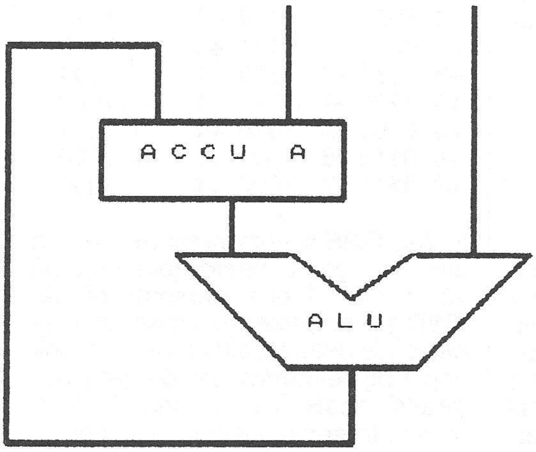
precies wat de uitdrukking **'64 KByte computer'** inhoudt.
Je moet daarbij wel in aanmerking nemen, dat dit niet altijd **gebruikersgeheugen** hoeft in te houden. De mooiste voorstelling van het geheugen is een ongelooflijk grote ladenkast die precies **64K** ofwel **65536** laatjes bevat, en in elk laatje past de digitale informatie van 1 Byte (8 bits). Adresseren we bijvoorbeeld adres **$E13C**, zoals we deden in het schema, dus 57660 decimaal, dan stellen we ons voor om laatje 57660 open te trekken en te kijken wat voor digitale informatie er in zit, bijvoorbeeld **%01011101**. Andersom kan het ook, door een laatje te adresseren, daarna open te trekken en er digitale informatie in te stoppen. Zo werkt in principe ook het echte computer-geheugen, alleen natuurlijk niet met laatjes. De bedoeling van dit verhaaltje was om ongeveer een indruk te krijgen hoe een geheugen in wezen werkt.

De ALU

Nu we de principes van de werking van het computergeheugen min of meer begrijpen, kunnen we verder gaan met de ALU, een ander onderdeel van de processor waar we veel mee te maken zullen krijgen. De ALU is een deel binnen in de microprocessor, die de rekenkundige handelingen uitvoert. **ALU** betekent **Arithmatic Logic Unit** wat eigenlijk zoiets als *'logische rekenkundige eenheid'* betekent, een beetje vrij vertaald overigens. Deze ALU wordt heel vaak gebruikt voor **berekeningen** maar kan ook eventueel worden ingezet als tijdelijk hulpregister voor het verplaatsen van gegevens of getallen binnen het computergeheugen.
Dus als je een getal of een letter ergens anders in het geheugen wil plaatsen dan kun je dat met hulp van de ALU doen, sterker nog, als je ook maar even met het getal wilt rekenen heb je de ALU echt nodig. De ALU beschikt daarnaast over een aantal hulpmiddeltjes voor de **manipulatie** met getallen en Bytes. Deze hulpmiddeltjes worden nu **FLAGS** genoemd en vertellen in veel gevallen wat er met een bewerking, een machinetaal-instructie dus, is gebeurd. Om dit een beetje duidelijker te maken gaan we eerst eens kijken wat de FLAGS van de ALU nu precies betekenen.

FLAG		Betekenis
C	=	**Carry bit**
Z	=	**Zero bit**
I	=	**Interrupt bit**
D	=	**Decimal mode bit**
B	=	**Break bit**
V	=	**Overflow bit**
N	=	**Negative bit**

In de tabel staan de zogeheten **FLAGS** of vlaggen van het rekenregister (ALU) afgebeeld. Deze FLAGS hebben geen enkele invloed op de directe **uitvoering** van het programma, ze geven alleen maar aan wat er **na een bewerking** is gebeurd. Het is dus aan de programmeur om deze FLAGS te gebruiken om een programma in goede banen te leiden. Een FLAG zou je in principe kunnen vergelijken met de **remlichten** van een auto. Als bij de chauffeur van de auto die voor je rijdt de remlichten gaan branden, dan betekent het dat die auto aan het remmen is. Negeer je deze waarschuwing dan zou dat best een aanrijding kunnen worden. Deze remlichten hoeven dus niet direct jouw manier van rijden te beïnvloeden, al is het wel verstandig er rekening mee te houden.
Dit is met de FLAGS ook het geval, hou je er geen rekening mee, dan zit de kans er in dat het fout gaat.
In bovenstaand figuur zie je een schematische afbeelding van de **ALU**, je ziet hier een blokje met de naam **ALU** en een blokje genaamd **ACCU**. Als je een getal uit het geheugen haalt om een berekening mee te maken, dan wordt dit getal eerst in de **ACCU** geplaatst. Deze **ACCU** heeft niets te maken met auto's of remlichten, maar is een afkorting van het engelse **'ACCUmulator**, wat **opteller** betekent. Vervolgens kun je met dat getal uit de ACCU een bewerking uitvoeren. Je wilt er bijvoorbeeld een getal bij op tellen wat ergens anders in het geheugen staat. Deze twee verschillende getallen worden allebei aangeboden op de **ALU** (arithmatic logic unit), en de bewerking die van te voren (door de programmeur) is gekozen, wordt uitgevoerd. De uitkomst van de ALU gaat direkt weer terug in de ACCU, gereed voor een volgende bewerking.
De **ALU** kan alleen maar **logische** bewerkingen uitvoeren, zoals **optellen, aftrekken, NOR, NAND, AND, OR en Exclusive OR.**



De ALU reken-eenheid

Vermenigvuldigen kan hij in het geheel niet, dit wordt echter door de ALU/processor opgelost door een getal zo vaak op te tellen als voor de vermenigvuldiging nodig is.
Dus 4 x 12 is voor de ALU hetzelfde als 4 keer het getal 12 bij elkaar optellen.
De **snelheid** waarmee dit optellen in de ALU plaatsvindt is gelukkig zo hoog dat dit de gebruiker nauwelijks opvalt. De ACCU is ook 8 bits breed dus het grootste getal wat in principe in de ACCU kan is **$FF** of **%11111111** of **255** decimaal (2 tot 8ste macht). Dit is een getal wat bij veel Basic-kenners een lichtje zal doen branden, omdat dit meestal de maximale waarde is als instelparameter bij bijvoorbeeld de geluidschip-POKEs en het maken van **sprites**.

Negatief

We zijn er tot nu toe voor het gemak maar even van uit gegaan dat we alleen met **positieve** getallen werken, wat echter natuurlijk niet altijd het geval is. Helaas krijgen wij ook vaak te maken met negatieve getallen in het **binaire stelsel**. Dit is door de heren ontwerpers opgelost door het **achtste bit** in de ACCU te gebruiken als *wijzer* voor een negatief of positief getal, namelijk een 0 staat voor positief en een 1 betekent negatief. Dit wordt toegepast in een paar ingewikkelde stelsels: het **one's** en het **two's complement** stelsel.
Dat heeft als bijwerking in machinetaal, dat het hoogste getal in de accu daardoor niet meer 255 is, maar slechts de helft, dus 127 (positief) en -128 (negatief). Dus als je bij het positieve getal 127, 1 optelt dan is de

uitkomst plotseling negatief geworden.
Dit zijn voor de beginner maar rare begrippen, waar we in deze aflevering enige verduidelijking in zullen brengen.
Dit achtste bit is in het schema terug te vinden als het NEGATIVE bit, dus zodra het achtste bit een 1 is, vat de processor dit als een negatief getal op. Het volgende verhaal is een stuk wat enige aandacht vereist, omdat je snel verward raakt met de begrippen **Overflow, Carry** en **Negative bit**. Het **Carry-bit** is één van de FLAGs die wordt 'gezet' als er sprake is van een **drager**. Als voorbeeld nemen we hiervoor twee decimale getallen 6 en 7 die we gaan optellen. De uitkomst is 13, en de 1 is hier de Carry (drager).

```
 1    1
  6   0101
  7   0011
----  -----
 13   1000
```

Het **Overflow-bit** wordt gezet als een getal niet meer in de daarvoor bedoelde ruimte past, dus als een uitkomst van een berekening bijvoorbeeld acht bits wordt en er past maar zeven bits in het register, naast het 8ste positief/negatief bit, dan worden de zeven bits gewoon in het register gezet, en het Overflow-bit wordt 'gezet' bij de FLAGs. We gaan hier straks wat mee rekenen om het geheel wat begrijpelijker te maken. Voor je zover bent moet je echter eerst nog het één en ander weten over het ONE's en TWO's complement.

ONE's en TWO's Complement

Het ONE's en TWO's complement bestaat uit twee stelsels waar je positieve van negatieve getallen kunt onderscheiden. Het TWO's complement is het stelsel waar we het meest mee rekenen, omdat dit stelsel de logica meer benadert dan het ONE's complement.
Voorbeelden van beide stelsels zullen dit, hoop ik, wel wat duidelijker maken. Zoals al eerder gezegd, wordt het positief of negatief zijn van een **Byte** of getal duidelijk gemaakt door het meest linkse, dus laatste of achtste bit 0 (positief) of 1 (negatief) te maken.
Door alleen deze handeling uit te voeren ben je er helaas nog niet, want ook de andere 7 bits moeten een soortgelijke behandeling ondergaan. Dit is voor het ONE's complement nu net weer iets anders dan voor het TWO's complement stelsel, gelukkig maar......

Hoewel het helemaal niet gemakkelijk is om deze stelsels snel en goed te kunnen begrijpen, zullen we toch een poging wagen , door aan de hand van een paar rekenvoorbeelden te laten zien wat de uitkomst wordt bij beide stelsels.

Omrekenen

Getal		One's C		Two's C	
+0	0000	-0	1111	+1	-/+0 0000
+1	0001	-1	1110	+1	-1 1111
+2	0010	-2	1101	+1	-2 1110
+3	0011	-3	1100	+1	-3 1101
+4	0100	-4	1011	+1	-4 1100
+5	0101	-5	1010	+1	-5 1011
+6	0110	-6	1001	+1	-6 1010
+7	0111	-7	1000	+1	-7 1001

In het **ONE's complement** worden alle bits geïnverteerd (omgedraaid van 0 naar 1 of andersom), bij het **TWO's complement** wordt precies hetzelfde gedaan, alleen wordt er ook nog 1 bij het geïnverteerde getal **opgeteld**, zoals het schema duidelijk maakt. In eerste instantie lijkt het rekenen met het **One's complement** eenvoudiger, maar wanneer je gaat optellen en aftrekken met getallen in One's en Two's complement getallen zal snel duidelijk zijn dat het rekenen met Two's complement getallen eenvoudiger en logischer is dan met het One's complement. Om deze reden rekenen we in het vervolg verder met het Two's complement getallenstelsel en vergeten we het One's complement stelsel, om niet nog veel meer in de war te raken.

One's complement

```
+12     00001100
 - 8    11110111
+----  ------------+
 +4   (1)00000011
          (1) Carry er bij op tellen
          (011 + 1 = 100)
        ------------
 +4     00000100
Teken positief dus niet omdraaien

-12     11110011
+ 8     00001000
+-----  ------------
 - 4  (1)1111011
Teken negatief dus inverteren
        ------------
      -(0)0000100
          Uitkomst is dus %0100
```

Two's complement

```
+12     00001100
 - 8    11111000
+-----  ------------
 + 4  (1)00000100

-12     11110100
 + 8    00001000
+-----  ------------
 - 4    11111100
```

Omdat we nu een beetje meer van ONE's en TWO's complement weten is het tijd geworden weer terug te komen op de FLAGs.

De FLAGS

We beginnen met de **Overflow-** en de **Carry-flag**, omdat deze het meest door de programmeurs worden verwisseld. *Dus rustig en goed lezen wat er nu volgt*. De Overflow-flag wordt gezet als een getal **niet** meer in het register van 8 bits past, dus stel dat een bijvoorbeeld een decimaal getal maar ten hoogste 3 cijfers mag bevatten en je hebt het getal 960 waar je 100 bij wilt optellen, dan kom je uit op 1060. Dit getal past niet in het register, omdat dat maar 3 cijfers breed kon zijn. In dit geval wordt bij de computer de **Overflow-flag** gezet. Een Overflow treedt nooit op als je negatief en een positief getal bijelkaar optelt, want als je een dergelijke optelling uitvoert, beide met getallen bijvoorbeeld onder de tien, dan kun je nooit een getal krijgen wat **groter** is dan 10 (twee cijfers).
We kunnen hierover natuurlijk lange theorieën gaan afsteken, maar een paar voorbeelden met wat uitleg zal begrijpelijker zijn. De breedte van de getallen is in dit geval dus zeven bits met daarbij het achtste bit als **teken bit**. Bijvoorbeeld:

+105 =%01101001 en
-105 =%10010111

Het komt dan ook vaak voor als je twee grote positieve getallen bij elkaar optelt, de uitkomst ogenschijnlijk negatief wordt. In dit voorbeeld spreken we over een **overflow**. Bijvoorbeeld **01011100 + 00101100 = 10001000**.
Aan de 1 vooraan links (het 8ste bit), wat overigens het **MSB** (most significant bit) wordt genoemd, kun je zien dat **10001000** negatief is gewor-

den, als uitkomst van een optelling van twee positieve getallen! Dit is dus een **overflow**, maar wat ook gebeurd is, is het zetten van het **negatief bit**, want de 65XX processor weet gewoon niet beter of dit is een negatief getal geworden. Dus elke keer als het achtste bit een 1 is denkt de 65XX dat het een negatief getal betreft.
De Carry-flag wordt gezet als er een drager plaatsvindt, het zogeheten 'één onthouden', bijvoorbeeld **0011 + 0001 = 0100**. Hier is een (over)drager geweest van het tweede naar het derde bit. Als er een **Overflow** optreedt krijg je in principe ook een **Carry**. Door een samenspel van deze twee flags kunnen we straks gelukkig ook veel grotere getallen optellen, maar dat komt wel als we wat meer van machinetaal weten.

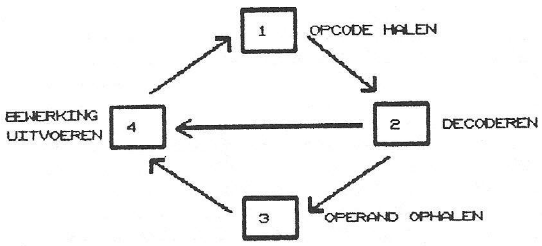
Reken voorbeelden

+85	**01010101**	
+45	**00101101**	
+----	**+ ------------**	
+130	**10000010**	**Overflow**
--109	**10010011**	
+ 31	**00011111**	
- -----	**- ------------**	
-140	**01110100**	**Overflow**
+54	**00110110**	
+39	**00100111**	
+ ----	**+ ------------**	
+93	**01011101**	**Carry**
+117	**01110101**	
+ 51	**00110011**	
- -----	**- ------------**	
+ 66	**01000010**	

Met deze rekenvoorbeelden moet je in staat kunnen zijn om enigszins te begrijpen wat het verschil is tussen **Overflow** en **Carry**. Het spitswerk met dit gereken komt hopelijk vanzelf als we later met de machinetaal-instructies aan het werk gaan. Til er voor nu maar niet te zwaar aan als je alles van dit rekenwerk niet meteen begrijpt.

Zero

Er blijft nu eigenlijk alleen het **Zero-bit** nog over van de FLAGs die voor ons belangrijk zijn. Deze **Zero-flag** wordt gezet als de uitkomst van een bewerking gelijk is aan 0. Dus als de Zeroflag 1 is, is de waarde in de ACCU 0. Als je 't eenmaal weet, vergeet je het nooit meer. Een eenvoudige FLAG waar we dan ook niet teveel woorden aan vuil zullen maken. Simpel gezegd: ***als het getal in ACCU-register = 0 dan is de Zero-FLAG 1 en omgekeerd.***
De **O- C- en Z-flag** zijn de drie Flags of vlaggen waar we in eerste instantie bij het programmeren in machinetaal mee te maken krijgen. Het diepere begrip van deze vlaggen komt dan ook vanzelf naar boven, als we maar voldoende oefenen in de loop van deze cursus.



Von Neumann's cyclus

Adresseer mogelijkheden

Voordat we met de eerste echte **instructie** beginnen moet je nog weten op welke manieren je kunt adresseren. In totaal kun je op ongeveer **13 verschillende manieren** adresseren, die we overigens niet allemaal in één keer gaan behandelen, om te voorkomen dat je na een half uur zweten de computer in de hoek gooit. We zullen beginnen met de begrippen '**Immediate**' en '**Absolute**', waar je eerst alleen al de nodige moeite mee zult hebben. Voordat we dit echt zullen uitleggen gaan we kijken hoe een microchip de instructies verwerkt.
De **instructie-cyclus** van Von Neumann geeft hierover wat duidelijkheid.
In dit voorbeeld zie je direct twee nieuwe woorden namelijk **opcode** en **operand**. Opcode betekent in wezen hetzelfde als **instructie**, wat in het Neumann-voorbeeld gewoon **Instructie halen** betekent. Operand is hetzelfde als de **te bewerken waarde**, wat niet altijd in elke machinetaal-instructie hoeft te zijn opgenomen.
Wat eigenlijk uit deze instructie-cyclus blijkt, is dat de microchip verschillende **manieren** van adresseren kent. Als jij als opdracht krijgt om een spijker in de muur te slaan, dan heb je daar op z'n minst een **spijker** voor nodig, wat in dit geval de **operand** zou kunnen zijn, terwijl de **hamer** als **opcode** zou kunnen dienen. Dus hier moet je niet alleen de instructie uitvoeren maar ook *'waarmee'* je de instructie moet uitvoeren. Je kunt -zoals gezegd- ook een instructie uitvoeren waar geen operand aan te pas komt, bijvoorbeeld als je met de hamer klopsignalen wilt geven. Wat het gebruik van de operand betreft hebben we nu direkt verteld wat de operand was (de spijker). De mogelijkheid die ook bestaat is dat je zegt '*Pak iets uit een bepaald bakje en sla dat in de muur*'. Op het moment weet je nog niet **wat** je in de muur moet slaan. Dat weet je pas als je **'het'** uit het bakje hebt gehaald.
Deze drie methodes hebben in de machinetaal een verschillende benaming namelijk:

IMMEDIATE,
IMPLIED en
ABSOLUTE.

Immediate noemen we ook de 2 Byte instructies zoals LDA #12, want de instructie neemt 1 Byte en de operand (waarmee je iets gaat doen) neemt ook 1 Byte, in totaal dus 2 Bytes.
Implied noemen we bijvoorbeeld de 1 Byte instructie TXA, want hier hoeft alleen de instructie uitgevoerd te worden, bijvoorbeeld het overbrengen van een dataByte van register X naar de ACCU.
En bij de **Absolute** adressing methode worden meestal 3 Byte's gebruikt, omdat naast de opcode van 1 Byte nog een compleet geheugenadres moet worden aangewezen om het te gebruiken of te bewerken deel te halen. STA $5324 en LDA $E23C zijn daar voorbeelden van. Dit adres is 16 bits dus 2 keer 1 Byte, vandaar de 3 Byte's voor het voltooien van 1 instructie.

Tenslotte

Dit waren alvast drie adressing methodes waar we in de volgende afleveringen verder op in zullen gaan, maar dan in machinetaal.
Alvast veel succes met deze aflevering, doe je best en tot de volgende keer.

Tjipke van der Land